New York IT Operations Management

Last updated: January 22, 2020
Some examples and graphics depicted herein are provided for illustration only. No real association or connection to ServiceNow products or services is intended or should be inferred.

If you have comments about this documentation, submit your feedback to:
docfeedback@servicenow.com
# Table of Contents

**IT Operations Management**

- Subscriptions for IT Operations Management
  - View subscription statistics for IT Operations Management
  - Check CI count used for IT Operations Management subscriptions
  - Check which CI types require an IT Operations Management subscription

**ITOM Visibility**

- Service Mapping
- Discovery
- Discovery patterns used by ITOM Visibility

**ITOM Health**

- Event Management
- Operational Intelligence

**ITOM Optimization**

- Cloud Management
- Cloud Optimization: Cloud Insights application

**Index**
## IT Operations Management

ServiceNow® IT Operations Management solutions help your organization enhance visibility into your infrastructure and services, prevent service outages, and maximize operational agility.

### Watch the video

### See more, know more, do more with IT Operations Management

Move IT operations from a reactive team to one that works intelligently for the business with IT Operations Management. Deliver high-performance business services. Gain visibility across infrastructure and apps, maintain service health, and optimize cloud delivery and spend.

<table>
<thead>
<tr>
<th>ITOM Visibility</th>
<th>ITOM Health</th>
<th>ITOM Optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Take charge of your infrastructure and services from a single source of truth.</td>
<td>Drive down service outages with AIOps.</td>
<td>Drive down cloud costs while enhancing the agility of your cloud.</td>
</tr>
</tbody>
</table>
ITOM Visibility

IaaS  PaaS  Serverless

amazon web services™
Microsoft Azure
Google Cloud
IBM Cloud

CMDB
ITOM Visibility gives you an accurate, up-to-date view of your IT infrastructure and services, spanning both multi-cloud and on-premises environments. It automates the end-to-end infrastructure discovery and service mapping process—including tracking ongoing changes—creating a complete and reliable record in your CMDB. This infrastructure and service information is seamlessly leveraged by other ServiceNow applications such as ITOM Health, ITOM Optimization, and Software Asset Management, and you can easily enrich it with additional configuration information by integrating with third-party applications and data sources.
**ITOM Health** uses the power of AIOps to turn the tidal wave of events into a trickle of actionable alerts—cutting through the noise, pinpointing service issues, and helping you to rapidly identify and remediate the root cause. Unlike legacy event management systems, which are static and rule based, ITOM Health applies machine learning and advanced analytics to correlate events, adapting automatically to rapidly evolving virtualized and cloud environments.

**ITOM Optimization**

![ITOM Optimization Diagram]

**ITOM Optimization** delivers the visibility and control you need while enhancing the agility of the cloud. It provides comprehensive visibility of cloud costs and usage, identifying and automating opportunities to optimize cloud spend. And, it also includes automated cloud provisioning capabilities, allowing you to establish an effective, agile cloud governance model.

**Get started**

- Work with an implementation specialist to achieve your desired business outcomes. To learn more, visit the [Customer Success Center](#).
- Just starting out with ServiceNow or brushing up on certain skills? Choose the training that’s best for you at the [Now Learning center](#).
Products

Read the product documentation:

- **ITOM Visibility**
  - Discovery
  - Service Mapping

- **ITOM Health**
  - Event Management
  - Operational Intelligence

- **ITOM Optimization**
  - Cloud Management
  - Cloud Insights

Subscriptions for IT Operations Management

The ServiceNow platform uses a new licensing method where your organization is charged for using the following IT Operations Management products: ServiceNow® ITOM Visibility product, ServiceNow® ITOM Health product, and ServiceNow® ITOM Optimization product.

For licensing purposes, CI types are assigned to CI categories. The customer contract for your organization determines the ratios of CI allocation for each CI category. Information from the customer contract is synchronized with the ServiceNow platform. There are the following CI categories:

- Containers
- PaaS
- Servers
- Unresolved monitored objects

**Note:** CI types are the same as CI classes.

ServiceNow charges for the following IT Operations Management applications:

**ITOM Visibility**

The ServiceNow® ITOM Visibility product consists of the ServiceNow® Discovery feature and the ServiceNow® Service Mapping feature, which together give you a unified, connected view of your entire IT network and the services it supports. For licensing purposes, ITOM Visibility provides the count of CIs that it discovers, both during horizontal and top-down discovery. ITOM Visibility reports the count for CIs of the following CI categories: Servers, PaaS, and Containers.

**ITOM Health**

ServiceNow® ITOM Health product includes the ServiceNow® Event Management feature and the ServiceNow® Operational Intelligence feature, which together help you to track and maintain the health of services in your organization. ITOM Health reports the number of CIs, for which it captures alerts, events, and metric data. ITOM Health reports the CI count for all CI categories.

**ITOM Optimization**
ServiceNow® ITOM Optimization product includes two features: The Cloud Management feature gives you tools to provision private and public cloud infrastructure and services and to achieve consistent management and cost visibility. The Cloud Insights application, available in the ServiceNow Store, helps you to analyze the full range of costs associated with cloud assets so you can identify and take action on opportunities to save money and optimize operations. For licensing purposes, ITOM Optimization provides the count of CIs of the following categories: Servers, Containers, and PaaS.

Depending on the needs of your organization, you can purchase subscriptions for each IT Operations Management product separately (a la carte) or together (in bundles). When you purchase subscriptions in bundles, you get the same number of subscriptions for all IT Operations Management applications covered by the bundle. For example, for a bundle of 500 that covers ITOM Visibility and ITOM Health, your organization receives 500 subscriptions for ITOM Visibility and 500 subscriptions for ITOM Health.

Bundle subscriptions cover specific applications. You cannot use bundle subscriptions for other applications, even if these other applications are part of the same bundle. For example, you purchased a bundle of 500 covering ITOM Visibility and ITOM Health, and you used up all 500 ITOM Visibility subscriptions. You cannot use the spare ITOM Health subscriptions for ITOM Visibility.

If your organization consumes more subscriptions than you initially estimated, you can purchase a larger bundle that better suits the needs of your organization. Your organization may consume subscriptions of IT Operations Management applications at different levels. If so, you can purchase a bundle that sufficiently covers most of the required applications and also purchase a la carte subscriptions for the application that consumes more subscriptions.

Note: IT Operations Management bundles also provide entitlement to IntegrationHub for the functionality related to IT Operations Management applications.

View subscription statistics for IT Operations Management

View how many subscriptions for IT Operations Management applications your organization purchased and consumed.

Role required: sn_itom_license.reader

Review the configuration item (CI) allocation and allocation level to see how your organization uses IT Operations Management subscriptions and plan upcoming subscription needs.

View the following statistics on subscriptions purchased a la carte and by bundles:

Name
The name of either subscription bundle or IT Operations Management application, if your organization purchased subscriptions per application separately (a la carte).

Purchased
The number of purchased subscriptions per bundle or application (a la carte).

Allocated
The number of consumed subscriptions from a bundle or application (a la carte). For bundles, this field shows the highest number of consumed subscriptions by applications that are part of the bundle. For example, if ITOM Visibility consumed 300 subscriptions and ITOM Health consumed 200 subscriptions, this field shows 300 for the bundle covering these applications.

The color code indicates the percentage of the subscriptions that your organization consumed. By default, the color code threshold is 90%.

- Green — Your organization has used less than 10% of purchased subscriptions.
- **Yellow** — Your organization has used more than 90%, but less than 100% of purchased subscriptions.
- **Red** — Your organization has used 100% or more and exceeded the number of purchased subscriptions. Purchased subscriptions are overdrawn.

**Start date/End date**

The dates for which this subscription is valid.

The licensing module calculates and displays subscription consumption as follows:

**Subscriptions by bundle only**

When you purchase subscriptions in bundles, you get the same number of subscriptions for all IT Operations Management applications covered by the bundle. For example, for a bundle of 500 that covers ITOM Visibility and ITOM Health, your organization receives 500 subscriptions for ITOM Visibility and 500 subscriptions for ITOM Health.

The licensing module subtracts the number of consumed subscriptions from the bundle subscriptions for the relevant application. Bundle subscriptions cover specific applications. You cannot use bundle subscriptions for other applications, even if these other applications are part of the same bundle. For example, you purchased a bundle of 500 covering ITOM Visibility and ITOM Health, and you used up all 500 ITOM Visibility subscriptions. You cannot use the spare ITOM Health subscriptions for ITOM Visibility.

If your organization exceeds the number of purchased subscriptions, the bundle size is automatically adjusted to the number of consumed subscriptions. When that happens, the licensing module recalculates levels of consumption for all applications covered by the same bundle.
The Subscriptions window displays the actual number of consumed subscriptions under **Allocated**. The red color code indicates that the bundle is overdrawn.

**Subscriptions a la carte only**

The licensing module subtracts the number of consumed subscriptions from the a la carte subscriptions for the relevant application. The Subscriptions window displays the information for purchased and allocated subscriptions for IT Operations Management applications.
If your organization exceeds the number of purchased subscriptions for an IT Operations Management application, you cannot use unconsumed subscriptions for another application.

<table>
<thead>
<tr>
<th>Subscriptions window</th>
<th>ITOM Health</th>
<th>Capacity</th>
<th>ServiceNow</th>
<th>Start date</th>
<th>End date</th>
<th>Display on</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ITOM Health</td>
<td></td>
<td>ServiceNow</td>
<td>2019-03-01</td>
<td>2019-06-28</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td>ITOM Visibility</td>
<td></td>
<td>ServiceNow</td>
<td>2019-03-01</td>
<td>2019-06-28</td>
<td>false</td>
</tr>
</tbody>
</table>

If your organization exceeds the number of subscriptions, the Subscriptions window shows the relevant a la carte subscription is overdrawn. In the following figure, ITOM Health consumed more subscriptions than you purchased for it.
Subscriptions for the same applications both in bundle and a la carte

If you purchased ITOM subscriptions both in bundle and a la carte, the licensing module always subtracts the number of consumed subscriptions from the bundle before deducting from the number of subscriptions purchased a la carte. For example, there is a bundle of 500 subscriptions covering ITOM Visibility and ITOM Health. In addition, there are 250 subscriptions for ITOM Visibility purchased a la carte. The first 500 subscriptions consumed by ITOM Visibility are consumed by the bundle. Only when ITOM Visibility exceeds the number of subscriptions in the bundle, the licensing module starts deducting from subscriptions purchased a la carte and shows them as subscriptions allocated to ITOM Visibility. In the following figure, one or both applications covered by the ITOM Pro bundle consumed 250 subscriptions.

If your organization exceeds the number of subscriptions purchased in bundle and a la carte, the licensing module considers it as overdrawn from the a la carte subscriptions.
In this case, the Subscriptions window indicates that the a la carte subscriptions are overdrawn.

Subscriptions in bundle and subscriptions a la carte for some applications covered by the bundle

If you purchased ITOM subscriptions in bundle and you also purchased a la carte subscriptions for an application that the bundle covers, the licensing module always subtracts the number of consumed subscriptions from the bundle before deducting from the number of subscriptions
purchased a la carte. In the following figure, an organization consumed all subscriptions available by the ITOM Pro bundle for ITOM Health. The licensing module then started subtracting from the a la carte subscriptions for ITOM Health.

If your organization exceeds the overall number of subscriptions both in the bundle and a la carte subscriptions, the licensing module considers it as overdrawn on the a la carte subscription. If your organization exceeds subscriptions for the application covered only by the same bundle, it is considered overdrawn on the bundle.

In this case, the Subscriptions window displays the a la carte subscriptions for this application as overdrawn. If your organization exceeds the overall number of subscriptions for an application
that is covered only by the bundle, the Subscriptions window shows the bundle as overdrawn. In the figure below, ITOM Health consumed 810 subscriptions going over subscriptions provided by the bundle (500) and a la carte (250). The Subscriptions window shows the ITOM Health a la carte subscription as overdrawn. ITOM Visibility consumed 550 subscriptions. Since there are no a la carte subscriptions purchased for ITOM Visibility, the bundle that covers ITOM Visibility appears overdrawn.

To view subscription information for IT Operations Management subscriptions purchased a la carte and in bundles, navigate to ITOM License > License Summary.

Use subscription names to differentiate between subscriptions bought in bundles and a la carte.

Check CI count used for IT Operations Management subscriptions

ITOM Visibility, ITOM Health, and ITOM Optimization provide information on licensable resources that these ITOM applications serve. Resources that IT Operations Management applications discover, monitor, and provision are configuration items (CIs) stored in the CMDB. The ITOM
licensing module combines this information on CIs with the information on subscriptions your organization purchased to produce statistics on subscription use by IT Operations Management applications.

Role required: admin

ServiceNow charges for the following IT Operations Management applications:

**ITOM Visibility**

The ServiceNow® ITOM Visibility product consists of the ServiceNow® Discovery feature and the ServiceNow® Service Mapping feature, which together give you a unified, connected view of your entire IT network and the services it supports. For licensing purposes, ITOM Visibility provides the count of CIs that it discovers, both during horizontal and top-down discovery. ITOM Visibility reports the count for CIs of the following CI categories: Servers, PaaS, and Containers.

**ITOM Health**

ServiceNow® ITOM Health product includes the ServiceNow® Event Management feature and the ServiceNow® Operational Intelligence feature, which together help you to track and maintain the health of services in your organization. ITOM Health reports the number of CIs, for which it captures alerts, events, and metric data. ITOM Health reports the CI count for all CI categories.

**ITOM Optimization**

ServiceNow® ITOM Optimization product includes two features: The Cloud Management feature gives you tools to provision private and public cloud infrastructure and services and to achieve consistent management and cost visibility. The Cloud Insights application, available in the ServiceNow Store, helps you to analyze the full range of costs associated with cloud assets so you can identify and take action on opportunities to save money and optimize operations. For licensing purposes, ITOM Optimization provides the count of CIs of the following categories: Servers, Containers, and PaaS.

The process of collecting and aggregating information for licensing purposes consists of the following interactions:

1. The IT Operations Management applications count CIs and assign them to CI categories daily. For more details, see KB0748149: CI count for licensing purposes by IT Operations Management.

2. The IT Operations Management applications daily report CI count by CI category to the IT Operations Management licensing module. If features of the same application report the same CIs as their resources, the CI count is manipulated to remove the duplication. For example, if Discovery and Service Mapping discover the same CIs, these CIs appear only once in the CI count.

3. The licensing module aggregates CI counts from IT Operations Management applications to get the daily average CI count for the last 90 days.

4. The licensing module correlates the daily average CI counts for IT Operations Management applications with the licensing information from the customer contract to produce license statistics.

As a result, you can view the statistics on how your organization uses purchased subscription units.
You can view the following information on CI count and subscriptions purchased for each IT Operations Management application separately (a la carte):

- **Total Count**: CI count for CIs organized by CI categories, for each IT Operations Management application.
- **Subscription Unit Ratio**: Ratios define how many CIs of a certain CI category require a subscription. The licensing module pulls the information about ratios from customer contracts.
- **Total Subscription Units Consumed**: The number of subscriptions per CI category for each IT Operations Management application that your organization consumed. The licensing module calculates this number by applying the subscriptions ratio to the CI count number for each application's CI category.
- **Total Subscription Units Consumed**: The sum of subscriptions consumed by all IT Operations Management applications.
This information does not include any statistics on subscriptions purchased in bundles. For complete information on subscriptions, view subscription statistics for IT Operations Management.

To view subscription information for each IT Operations Management application separately (a la carte), navigate to ITOM License > License Report.
### Value Stream: Health (4)

<table>
<thead>
<tr>
<th>CI Category</th>
<th>Total Count</th>
<th>Subscription Unit Ratio</th>
<th>Total Subscription Units Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Containers</td>
<td>91</td>
<td>1:1</td>
<td>30</td>
</tr>
<tr>
<td>PaaS</td>
<td>2</td>
<td>1:1</td>
<td>0</td>
</tr>
<tr>
<td>Servers</td>
<td>958</td>
<td>1:1</td>
<td>958</td>
</tr>
<tr>
<td>Unresolved monitored objects</td>
<td>445</td>
<td>1:1</td>
<td>445</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td><strong>1,433</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Value Stream: Optimization (4)

<table>
<thead>
<tr>
<th>CI Category</th>
<th>Total Count</th>
<th>Subscription Unit Ratio</th>
<th>Total Subscription Units Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Containers</td>
<td>135</td>
<td>1:1</td>
<td>45</td>
</tr>
<tr>
<td>PaaS</td>
<td>4</td>
<td>1:1</td>
<td>1</td>
</tr>
<tr>
<td>Servers</td>
<td>135</td>
<td>1:1</td>
<td>135</td>
</tr>
<tr>
<td>Unresolved monitored objects</td>
<td>18</td>
<td>1:1</td>
<td>18</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td><strong>199</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Value Stream: Visibility (4)

<table>
<thead>
<tr>
<th>CI Category</th>
<th>Total Count</th>
<th>Subscription Unit Ratio</th>
<th>Total Subscription Units Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Containers</td>
<td>301</td>
<td>1:1</td>
<td>100</td>
</tr>
<tr>
<td>PaaS</td>
<td>18</td>
<td>1:1</td>
<td>6</td>
</tr>
<tr>
<td>Servers</td>
<td>773</td>
<td>1:1</td>
<td>773</td>
</tr>
<tr>
<td>Unresolved monitored objects</td>
<td>13</td>
<td>1:1</td>
<td>13</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td><strong>892</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Check which CI types require an IT Operations Management subscription

View the list of configuration item (CI) types that require IT Operations Management subscriptions.

Role required:

sn_itom_license.reader

For licensing purposes, CI types are assigned to CI categories. The customer contract for your organization determines the ratios of CI allocation for each CI category. Information from the customer contract is synchronized with the ServiceNow platform. There are the following CI categories:

- Containers
- PaaS
- Servers
- Unresolved monitored objects

**Note:** CI types are the same as CI classes.

The system stores information about CI categories and CI types that belong to them in the ITOM Licensing MetaData (itom_lu_metadata) table. This table includes information only about parent CI types. All CI types that are created by extending these parent CI types are included by default. For example, the Apache Web Server CI type is included in the CI count, because it is an extension of the Web Server CI type.

1. Navigate to ITOM License > Licenses by CI Type.
2. Click the arrow next to the CI Category to expand it and view CI types that are associated with it.
   
   CI types that appear under CI categories on this list require IT Operations Management subscriptions.

**ITOM Visibility**

The ServiceNow® ITOM Visibility product consists of the ServiceNow® Discovery feature and the ServiceNow® Service Mapping feature, which together give you a unified, connected view of your entire IT network and the services it supports.

**ITOM Visibility**

The ServiceNow® ITOM Visibility product consists of the ServiceNow® Discovery feature and the ServiceNow® Service Mapping feature, which together give you a unified, connected view of your entire IT network and the services it supports.

Discovery provides a replicable and reliable method to identify your enterprise IT infrastructure. Service Mapping creates an accurate, service-aware view of infrastructure. Service Mapping continuously monitors IT resources for service-affecting changes and updates service maps in real time.

For more information about ITOM Visibility, see the ITOM Visibility release notes.

**Using guided setup to implement IT Operations Management applications**

IT Operations Management Guided Setup provides a sequence of tasks that help you configure IT Operations Management applications on your ServiceNow instance. To open IT Operations...
Management guided setup, navigate to Guided Setup > ITOM Guided Setup. For more information about using the guided setup interface, see.

**Service Mapping**

The ServiceNow® Service Mapping application discovers all application services in your organization and builds a comprehensive map of all devices, applications, and configuration profiles used in these business services. Service Mapping maps dependencies, based on a connection between devices and applications. This method is referred to as top-down mapping. The top-down mapping helps you immediately see the impact of a problematic object on the rest of the application service operation.

Service Mapping is available as a separate subscription and requires activation by Service Mapping personnel.

**Explore**
- Upgrade to New York
- Understanding Service Mapping
- Applications supported by Discovery and Service Mapping
- Service Mapping roles and flow
- Domain separation and Service Mapping
- Subscriptions for IT Operations Management

**Set up**
- Service Mapping setup
- Request Service Mapping
- Advanced Service Mapping configuration

**Administer**
- Application service mapping
- Fix application service errors in bulk
- Fix errors in individual application service maps
- Review and approval of application service maps
- Video: Advanced Service Mapping features

**Use**
- Application service analysis and maintenance using maps
- Video: Service Mapping basics

**Develop**
- Discovery patterns used by ITOM Visibility
- Third-party service import
- Developer training
- Developer documentation

**Troubleshoot and get help**
- Ask or answer questions in the ITOM forum
- Resolve pattern-related mapping errors
- Search the HI knowledge base for known error articles
- Contact ServiceNow Technical Support

**Understanding Service Mapping**

Service Mapping discovers all application services in your organization and builds a comprehensive map of all devices, applications, and configuration profiles used in these application services.

**Who uses Service Mapping?**

Service Mapping enables IT departments of companies, organizations, and cloud companies providing platform as a service to create a service-aware view of infrastructure.
How do you use Service Mapping?

An application service is a set of interconnected applications and hosts which are configured to offer a service to the organization. Application services can be internal, like an organization email system or customer-facing, like an organization website. For example, creating financial reports through a web-based application requires a computer, web server, application server, databases, middleware, and network infrastructure. These applications and hosts are all configured to offer the service of financial reporting.

Typically, IT departments create and maintain an inventory that treats devices and applications as standalone, independent objects. Connections between the devices and applications are not included. This is usually referred to as horizontal discovery. This method does not address the biggest challenge for IT departments, which is understanding the connection and dependencies between each object.

Service Mapping maps dependencies, based on a connection between devices and applications. This method is referred to as top-down mapping. The top-down mapping helps you immediately see the impact of a problematic object on the rest of the application service operation.

Comparison of horizontal and top-down mapping results

Application service maps show infrastructure objects and semantic connections between them. Service Mapping regenerates application service maps regularly, to keep them updated and relevant. Any faulty objects are shown along with the devices and applications they affect, providing a visual clue of the state of the application service.
How does Service Mapping work?

The main method of Service Mapping discovering and mapping devices and applications is using patterns. A pattern is a sequence of operations whose purpose is to detect attributes of devices and applications and their outbound connections. Service Mapping can also discover devices and applications by following traffic connections between them. This method is referred to as traffic-based discovery.

What to know before you begin?

You can use Service Mapping only if the Discovery product is activated and set up. You must define users and configure credentials to allow Service Mapping and Discovery access to applications and devices inside your organization network. For more information, see Service Mapping setup.

Service Mapping on the Now Platform

Service Mapping relies on Discovery and the MID Server to discover devices and applications. Service Mapping uses results of horizontal discovery performed by Discovery. The MID Server facilitates communication between Service Mapping and devices and applications it discovers. Data collected and organized by Service Mapping is visible in Event Management, Dependency Views, and Application Portfolio Management (APM). With Event Management, you can view events to take actions for recovering your organization application services. Dependency Views shows relationships between devices and applications in the context of application services they belong to. When integrated with APM, Service Mapping provides information about components making up a business application and helps APM users to monitor business application performance.

Service Mapping supports domain separation. If your Now Platform uses domain separation, administrators and users can only see and manage application services belonging to their own domain.

Pattern-based discovery in Service Mapping

Pattern-based discovery is the main method of Service Mapping collecting data about devices and applications used in application services. After Service Mapping collects data, it then creates a map of application services and stores the collected data in the CMDB. ServiceNow applications refer to devices and applications that comprise an application service as configuration items (CIs).

Service Mapping uses patterns to discover and map CIs. A pattern is a sequence of steps whose purpose is to detect attributes of a CI and its outbound connections. A typical Service Mapping pattern consists of two types of algorithms for identifying CIs and finding CI connections.

The starting point of any discovery process is an entry point. An entry point is how clients access an application service. For example, to map your electronic mailing application service, define an IP address or host name of the email server as an entry point. The discovery and mapping process begins from Discovery performing the horizontal discovery to identify the host. Once the host discovery is complete, Service Mapping starts the top-down discovery to find and map applications running on this host.

Service Mapping uses MID Servers to communicate with CIs in your organization. MID Servers are located inside your organization network and Service Mapping can communicate with them without traversing firewalls.
The discovery and mapping process consists of the following interactions:

1. A user defines an application service with an entry point for an application CI.
   Service Mapping creates a record for the new application service in the Application Service (cmdb_ci_service_discovered) table.

2. The device hosting the application is identified.
   a. Service Mapping checks if the device hosting this application CI exists in the CMDB.
   b. If the device hosting this application CI does not exist, Service Mapping triggers Discovery to detect host.
   c. Discovery creates the first set of probes for port discovery, referred to as Shazzam probes, and places them as a discovery request in the External Communication Channel (ECC) queue.
   d. The MID Server checks the ECC queue and retrieves the discovery request assigned to it.
   e. The MID Server runs the probes against the host and discovers open ports.
f. The MID Server passes information on the host ports to the ECC queue.
g. Discovery checks the ECC queue and receives information on the host ports.
h. These steps are repeated for other types of probes: classification, identification, and exploration.

i. Discovery adds the host to the CMDB.

j. During the host discovery using probes, Service Mapping checks the ECC queue if this process is complete. When the host discovery is complete, Service Mapping checks whether this host exists in the CMDB.

**Note:** For the detailed description of the horizontal discovery flow, refer to [Horizontal discovery process flow with probes and sensors](#).

3. Once the host is found in the CMDB, Service Mapping discovers the application running on this host.

   a. Service Mapping creates an application discovery request for the IP address of the entry point. It then writes the request in the ECC queue and assigns a MID Server to the request.

   b. The MID Server checks the ECC queue and retrieves the discovery request assigned to it.

   c. The MID Server starts running identification sections of the patterns associated with the classifier to find the match for the entry point. When the identification section matches the entry point, the pattern discovers a CI.
d. The MID Server starts running connectivity sections of the pattern to find outgoing connections of the newly discovered CI.

e. The MID Server passes information on the discovered CI, its attributes, and connections to the ECC queue.

f. Service Mapping checks the ECC queue and receives information on the newly discovered CI.

Whenever Service Mapping checks the ECC queue and receives information on a discovered CI, it checks these tables for any data on outbound connections related to the CI: the cmdb_tcp and sa_flow_connection tables. If these two tables contain unique data that patterns did not discover, Service Mapping enriches the information about the CI connections and adds them to the map.

g. Service Mapping writes the information into the CMDB and adds this CI to the application service map.
h. Service Mapping creates the discovery requests for all applications to which the newly discovered CI connects. Mapping is complete after Service Mapping maps a CI that either does not have any outbound connections or is marked as a boundary. A boundary makes Service Mapping stop discovery from this point and not follow outgoing connections.

4. The system regularly runs the service recomputation to query the CMDB for the latest CI changes, add data for network and storage paths, and apply CI impact rules.

Traffic-based discovery in Service Mapping

Service Mapping uses traffic-based connections to collect network statistics and perform traffic-based discovery. Not only can Service Mapping discover CIs using patterns, it can also discover them by following traffic connections between CIs. This method is referred to as traffic-based discovery.

Using traffic-based discovery is like casting a finer net, allowing Service Mapping to find even those CIs that it failed to discover using patterns. The advantage is discovering more CIs, at the same time this method may clutter an application service with irrelevant CIs. Typically, you use traffic-based discovery at the initial stages of discovering an application service and disable it once you completed discovery and fine-tuned the application service.

While using traffic-based discovery creates a more inclusive map, it may also result in mapping many redundant CIs that do not influence the application service operation. You can choose to hide CIs discovered using this method from an application service map. In that case, these CIs are still part of the application service, but they are not on its map.

The system uses commands and network flow logs to collect traffic-related data and saves them in the CMDB tables. Service Mapping retrieves this data from the tables to detect CI inbound and outbound connections.
### Tables containing data collected using traffic-based methods

<table>
<thead>
<tr>
<th>Table</th>
<th>Source</th>
<th>Used by Service Mapping to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow Connector (sa_flow_connection)</td>
<td>Netflow and VPC logs</td>
<td>• Discover dependencies, add connections during top-down discovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Detect possible entry points for application services and create application service candidates for mapping based on these entry points.</td>
</tr>
<tr>
<td>Flow Services IP/Port and Statistics (sa_flow_service)</td>
<td>Netflow and VPC logs</td>
<td>Discover all services listening on ports. In base system, Service Mapping does not use data from this table.</td>
</tr>
<tr>
<td>Flow Server Communication (sa_flow_server_comm)</td>
<td>Netflow and VPC logs</td>
<td>Discover services communicating to other services. In base system, Service Mapping does not use data from this table.</td>
</tr>
<tr>
<td>TCP Connection (cmdb_tcp)</td>
<td>netstat and lsof commands</td>
<td>Discover connections during top-down discovery.</td>
</tr>
</tbody>
</table>

In base systems, traffic-based discovery uses only TCP-related data collected with the help of the netstat and lsof commands. Discovery based on Netflow and VPC logs requires additional configuration. You can enrich your traffic-based discovery by configuring Service Mapping to perform data collection using Netflow and VPC logs. In addition, Service Mapping has access to TCP connection data collected by enhanced traffic-based horizontal discovery performed by Discovery.

By default, traffic-based discovery using commands is available in Service Mapping allowing it to use this method at all levels. You can enable traffic-based discovery at different levels listed here from the most global to the most specific:

**Product level**

In the base system, traffic-based discovery in Service Mapping is enabled. If necessary, you can fine-tune or disable traffic-based discovery at the product as described in Properties installed with Service Mapping under Components installed with Service Mapping.

**Application service level**

You can enable traffic-based discovery for a specific application service. In this case, Service Mapping uses this method for all CIs making up this application service, unless traffic-based discovery is disabled for some CI types or specific CIs.

**CI type level**

You can create a discovery rule to include or exclude a CI type from traffic-based discovery. This rule prevails over the setting you choose for an application service.

**Specific CI level**

You can create a discovery rule to include or exclude a specific CI from traffic-based discovery. This rule prevails over the setting you choose for an application service.

Rules for specific CIs take precedence over rules for CI types. For example, if you do not want to use traffic-based discovery on any Tomcat servers, you can define a CI type rule disabling the...
traffic-based discovery on the Tomcat table. At the same time, you can create a discovery rule enabling the traffic-based discovery for a specific Tomcat server. In that case, Service Mapping uses the traffic-based discovery only for this specific Tomcat server out of all Tomcat servers.

As traffic-based discovery method may clutter an application service with irrelevant CIs, Service Mapping uses an algorithm to reduce the number of erroneously mapped CI. You can further adjust traffic-based discovery to remove unwanted CIs as described in Remove CIs not belonging to application services.

Domain separation and Service Mapping

This is an overview of domain separation and Service Mapping. Domain separation enables you to separate data, processes, and administrative tasks into logical groupings called domains. You can then control several aspects of this separation, including which users can see and access data.

Overview

Support: Level 1

Domain separation is supported in this application. Not all ServiceNow applications support domain separation; some include limitations on the data and administrative settings that can be domain separated. To learn more, see Application support for domain separation.

In ServiceNow environments where domain separation is configured, customers can view or manage only application services in domains to which customers have access.
How domain separation works in Service Mapping

In domain-separated environments, switch into relevant leaf domains to access actions related to mapping performed by Service Mapping. A leaf domain is a child domain of the lowest level, meaning it does not have any child domains itself.

Domain separation is handled in a slightly different way than simply configuring separate domains. Notice that you must select or specify the relevant leaf domain to perform the following mapping actions in domain-separated environments:

- When creating an application service, the application service is assigned to the user's domain.
- When manually adding a CI to an application service, you can choose only CIs that belong to the service domain.
- When manually adding connections to CIs to an application service, users must belong to the same domain as the application service.
- When transferring service definitions between instances, imported application services and service groups are assigned to the user's domain.

Refer to Discovery patterns used by ITOM Visibility topic and MID Server configuration for Service Mapping for more context. The following topics provide operational information on how to use Service Mapping in deployments with domain separation:

- Enable traffic-based discovery for CI types or specific CIs
- Create or modify patterns
- Create entry point types for Service Mapping
- Enhance patterns without changing their identification sections
- Customize pattern operations

Discovery of application services on cloud using Service Mapping

Service Mapping can discover application services consisting of devices and applications deployed in cloud.

There are two types of cloud computing methods your enterprise may use: infrastructure as a service (IaaS) and platform as a service (PaaS). Service Mapping discovers application services based on IaaS in the same way as any application service based on infrastructure located inside an enterprise private network. It relies on Discovery to find hosts before discovering applications and connections between them using patterns. For more information on this discovery flow, see Pattern-based discovery in Service Mapping.

Discovering application services entirely or partially based on PaaS requires a different paradigm. Applications in such deployments are hosted by network devices on cloud. To discover an application service in such a deployment, you use Discovery and Service Mapping as follows:

- You configure a service account for cloud-computing platform hosting the application service. Service Mapping and Discovery use this service account for accessing devices and applications on this platform.
- Discovery performs the horizontal discovery using probes and finds the infrastructure devices in the cloud.
- Discovery performs the horizontal discovery using patterns and finds applications and services hosted on the cloud.
- Service Mapping performs the top-down discovery to find connections between devices and applications and build the map of the application service.
Applications supported by Discovery and Service Mapping

Discovery and Service Mapping can discover a wide range of operating systems and applications.

Discovery finds computers, servers, printers, a variety of IP-enabled devices, and the applications that run on them. It can then update the CIs in your CMDB with the data it collects. This discovery method is referred to as horizontal discovery. Service Mapping maps dependencies, based on a connection between devices and applications. This method is referred to as top-down mapping. The top-down mapping helps you immediately see the impact of a problematic object on the rest of the application service operation.

On top of hosts and applications supported by default, you can discover additional hosts and applications by deploying patterns available on Store.

If your organization uses devices or applications, which are not supported by default or using patterns available at Store, you can configure Discovery and Service Mapping to discover them as described in Discovery patterns used by ITOM Visibility.

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>.Net Framework</td>
<td>Windows</td>
<td>1.x.x,2.x.x,3.x.x,4.x.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>.NET Application</td>
</tr>
<tr>
<td>Advanced Business Application Programming (ABAP) SAP Central Services (ASCS)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP ASCS Application patterns</td>
</tr>
<tr>
<td>Adobe JRun</td>
<td>Windows/Unix</td>
<td>4.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Jrun</td>
</tr>
<tr>
<td>Alteon RadWare ADC</td>
<td>Alteon</td>
<td>v31, v29.5</td>
<td>SNMP</td>
<td>Top-down and horizontal</td>
<td>Alteon Load Balancer</td>
</tr>
<tr>
<td>Amazon Relational Database Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Amazon AWS Relational Database Service</td>
</tr>
<tr>
<td>Amazon Route53 Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Horizontal</td>
<td>Amazon AWS Route53 HD</td>
</tr>
<tr>
<td>Amazon Application Load Balancer Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down</td>
<td>Amazon AWS application ELB service - TD</td>
</tr>
<tr>
<td>Amazon Elastic Load Balancer Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Horizontal</td>
<td>Amazon AWS classic ELB Service - TD</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>------------------</td>
<td>---------</td>
<td>----------------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>Apache HBase</td>
<td>Unix</td>
<td>N/A</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Probes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• HBase - Version</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• HBase - Main Class</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• HBase - Configuration</td>
</tr>
<tr>
<td>Apache HTTP Server</td>
<td>Windows/Unix</td>
<td>2.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Apache</td>
</tr>
<tr>
<td>Apache Tomcat Servlet container HTTP web server</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Tomcat</td>
</tr>
<tr>
<td>Azure Database Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Azure Database</td>
</tr>
<tr>
<td>Azure DNS Zones Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Horizontal</td>
<td>Azure DNS Zones HD</td>
</tr>
<tr>
<td>Azure LoadBalancer Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Azure LoadBalancer TD</td>
</tr>
<tr>
<td>Azure WebSite Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down</td>
<td>Azure WebSite</td>
</tr>
<tr>
<td>BMC Control-M</td>
<td>Unix</td>
<td>6.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Cntf-M Enterprise Manager</td>
</tr>
<tr>
<td>CA eTrust Directory server</td>
<td>Windows</td>
<td>7.x, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>CA eTrust Directory server</td>
</tr>
<tr>
<td>CA Identity Manager Provisioning Server</td>
<td>Windows</td>
<td>11.x, 12.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>CA Identity Manager Provisioning Server for Windows</td>
</tr>
<tr>
<td>CA Introscope Enterprise Manager</td>
<td>Windows/Unix</td>
<td>8.x, 9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>CA Introscope Enterprise Manager</td>
</tr>
<tr>
<td>CA Policy Server</td>
<td>Windows/Unix</td>
<td>11.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>CA Policy Server</td>
</tr>
<tr>
<td>CA Site Minder Agent</td>
<td>Windows/Unix</td>
<td>11.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>CA Site Minder Agent</td>
</tr>
<tr>
<td>Cisco Content Services Switch Load balancer</td>
<td>ACE</td>
<td>6.x</td>
<td>SNMP/SSH</td>
<td>Top-down and horizontal</td>
<td>Cisco CSS SNP</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>------------</td>
<td>---------</td>
<td>----------------</td>
<td>-----------------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Cisco ACE Application Control Engine</td>
<td>ACE</td>
<td>2.x</td>
<td>SSH</td>
<td>Top-down and horizontal</td>
<td>ACE Load Balancer</td>
</tr>
<tr>
<td>Cisco Call Manager</td>
<td>Cisco</td>
<td>3.x, 4.x</td>
<td>SNMP/SSH</td>
<td>Top-down</td>
<td>Cisco Call Manager</td>
</tr>
<tr>
<td>Cisco Fiber Connect</td>
<td>Cisco</td>
<td>No data</td>
<td>SNMP/SSH</td>
<td>Top-down</td>
<td>Cisco Fibre Connect</td>
</tr>
<tr>
<td>Cisco Global Site Selector Load Balancer</td>
<td>ACE</td>
<td>3.x</td>
<td>SNMP/SSH</td>
<td>Top-down and horizontal</td>
<td>Cisco GSS</td>
</tr>
<tr>
<td>Cisco Unified Computing System</td>
<td>UCS</td>
<td>3.x</td>
<td>REST</td>
<td>Horizontal</td>
<td>UCS - HD</td>
</tr>
<tr>
<td>Citrix Delivery Controller</td>
<td>Windows</td>
<td>7.x&gt;7.5, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Delivery Controller</td>
</tr>
<tr>
<td>Citrix Usage Collector</td>
<td>Windows</td>
<td>7.x&gt;7.5, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Citrix Collector</td>
</tr>
<tr>
<td>Citrix Netscaler Global Server Load Balancing (GSLB)</td>
<td>Netscaler</td>
<td>10.x</td>
<td>SNMP/SSH/NS.conf</td>
<td>Top-down and horizontal</td>
<td>Netscaler GLB</td>
</tr>
<tr>
<td>Citrix License Server</td>
<td>Windows</td>
<td>7.x</td>
<td>WMI</td>
<td>Horizontal</td>
<td>License Server</td>
</tr>
<tr>
<td>Citrix Netscaler Load Balancer</td>
<td>Netscaler</td>
<td>9.x, 10.x</td>
<td>SNMP/SSH/NS.conf</td>
<td>Top-down and horizontal</td>
<td>Citrix Netscaler</td>
</tr>
<tr>
<td>Citrix Presentation Server</td>
<td>Windows</td>
<td>4.x, 5.x, 6.x&lt;6.5</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Citrix XenApp or Presentation Server</td>
</tr>
<tr>
<td>Citrix XenApp Server</td>
<td>Windows</td>
<td>6.x&gt;6.5, 7.x&lt;7.6</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Citrix XenApp or Presentation Server</td>
</tr>
<tr>
<td>Connect APK</td>
<td>Windows</td>
<td>1.x.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Connect-It Service</td>
</tr>
<tr>
<td>Docker Container Linux</td>
<td>Docker</td>
<td>No data</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Docker Pattern</td>
</tr>
<tr>
<td>Epic Systems Corporation</td>
<td>Unix</td>
<td>2014.x.x, 2015.x.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>EPIC Cache</td>
</tr>
<tr>
<td>F5 BIG-IP Device Service Clustering</td>
<td>F5</td>
<td>11.x, 12.x</td>
<td>SNMP/SSH/REST</td>
<td>Horizontal</td>
<td>F5 Cluster</td>
</tr>
<tr>
<td>F5 BIG-IP Load balancer</td>
<td>F5</td>
<td>11.x, 12.x</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>F5 Load Balancer</td>
</tr>
<tr>
<td>HAProxy Server</td>
<td>Unix</td>
<td>1.5, 1.6, 1.7</td>
<td>SSH</td>
<td>Top-down</td>
<td>HA Proxy</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------</td>
<td>---------------</td>
<td>-------------------</td>
<td>----------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>HP Operations Manager</td>
<td>Windows/Unix</td>
<td>9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>HP Operations Manager</td>
</tr>
<tr>
<td>HP Quality Center</td>
<td>Windows</td>
<td>10.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>HP Quality Center on Windows</td>
</tr>
<tr>
<td>HP Service Manager software</td>
<td>Windows/Unix</td>
<td>9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>HP Service Manager Application Server</td>
</tr>
<tr>
<td>HP UCMDB</td>
<td>Windows</td>
<td>No data</td>
<td>WMI/WinRM</td>
<td></td>
<td>HP UCMDB on Windows</td>
</tr>
<tr>
<td>HP-UX Unix Server</td>
<td>HPUX</td>
<td>11i</td>
<td>SSH</td>
<td>Horizontal</td>
<td>HP-UX Server</td>
</tr>
<tr>
<td>IBM AIX</td>
<td>Unix</td>
<td>6.x, 7.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>AIX Server</td>
</tr>
<tr>
<td>IBM CICS Transaction Gateway</td>
<td>Windows/Unix</td>
<td>5.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>CTG</td>
</tr>
<tr>
<td>IBM Customer Information Control System</td>
<td>Windows/Unix</td>
<td>4.x, 5.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>CICS</td>
</tr>
<tr>
<td>IBM DB2</td>
<td>Windows/Unix</td>
<td>8.x, 9.x, 10.x, 11.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>DB2</td>
</tr>
<tr>
<td>IBM Security Access Manager</td>
<td>ISAM</td>
<td>8.0.1</td>
<td>SNMP/REST</td>
<td>Top-down and horizontal</td>
<td>ISAM Server</td>
</tr>
<tr>
<td>IBM Tivoli Access Manager</td>
<td>Windows/Unix</td>
<td>6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>ITAM - Asset Center</td>
</tr>
<tr>
<td>IBM Tivoli WebSEAL</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Webseal</td>
</tr>
<tr>
<td>IBM WebSphere Application server</td>
<td>Windows/Unix</td>
<td>5.x, 6.x, 7.x, 8.x, 9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Websphere</td>
</tr>
<tr>
<td>IBM WebSphere DataPower SOA Appliance</td>
<td>DataPower</td>
<td>140, 150</td>
<td>SNMP/REST</td>
<td>Top-down and horizontal</td>
<td>DataPower</td>
</tr>
<tr>
<td>IBM WebSphere Message Queue</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x, 9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>WNQ</td>
</tr>
<tr>
<td>IBM WebSphere Portal</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Websphere Portal On Windows</td>
</tr>
<tr>
<td>IBM Z mainframe</td>
<td>Z/OS</td>
<td>OS/390</td>
<td>SSH</td>
<td>Horizontal</td>
<td>IBM zOS Server</td>
</tr>
<tr>
<td>Interconnect Web Service</td>
<td>Windows</td>
<td>2014</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Inter Connect</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>----------</td>
<td>---------------</td>
<td>-------------------</td>
<td>---------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>JBoss Application Server</td>
<td>Windows/Unix</td>
<td>6.x, 7.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Jboss pattern</td>
</tr>
<tr>
<td>Linux Server</td>
<td>Unix</td>
<td>Ubuntu 16, RHEL 5.x, CentOS 6.x, 7.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Linux Server</td>
</tr>
<tr>
<td>Microsoft Active Directory Domain Services</td>
<td>Windows</td>
<td>13, 30, 31, 44, 47, 52, 56, 69</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Active Directory Service pattern</td>
</tr>
<tr>
<td>Microsoft Azure SQL Database</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Cloud Database Instance</td>
</tr>
<tr>
<td>Microsoft Fast Search</td>
<td>Windows</td>
<td>2005</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Fast Search</td>
</tr>
<tr>
<td>Microsoft Identity Integration Server</td>
<td>Windows</td>
<td>2003</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>IIFP On Windows Pattern</td>
</tr>
<tr>
<td>Microsoft Internet Information Services</td>
<td>Windows</td>
<td>6.0, 7.x, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down and horizontal</td>
<td>IIS</td>
</tr>
<tr>
<td>Microsoft Message Queuing</td>
<td>Windows</td>
<td>3.x, 4.x, 5.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>MSMQ pattern</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MongoDB</td>
<td>Windows/Unix</td>
<td>3.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>mongos server</td>
</tr>
<tr>
<td>Netapp Storage</td>
<td>Netapp</td>
<td>7.x, 8.x</td>
<td>SNMP/Basic Auth</td>
<td>Horizontal</td>
<td>NetApp Storage 7-Mode</td>
</tr>
<tr>
<td>Nginx Web server, proxy server</td>
<td>Windows/Unix</td>
<td>1.10, 1.10.1</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Nginx Pattern</td>
</tr>
<tr>
<td>OpenText Documentum</td>
<td>Windows/Unix</td>
<td>6.x, 7.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Pattern for license server</td>
</tr>
<tr>
<td>Oracle Application Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Oracle iAS</td>
</tr>
<tr>
<td>Oracle Concurrent Server</td>
<td>Windows/Unix</td>
<td>10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Concurrent Server</td>
</tr>
<tr>
<td>Oracle Clusterware</td>
<td>Linux</td>
<td>10.x, 11.x, 12.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Oracle CRS</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>Windows/Unix</td>
<td>8.x.x, 9.x.x, 10.x.x, 11.x.x, 12.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Oracle DB</td>
</tr>
<tr>
<td>Oracle Database Advanced Queueing</td>
<td>Windows/Unix</td>
<td>9.x.x, 10.x.x, 11.x.x, 12.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Advanced Queue Queue</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------</td>
<td>------------------</td>
<td>-------------------</td>
<td>----------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Oracle Discover Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Discoverer Engine</td>
</tr>
<tr>
<td>Oracle Forms</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Forms Engine</td>
</tr>
<tr>
<td>Oracle Fulfillment Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Fulfillment Server</td>
</tr>
<tr>
<td>Oracle GlassFish Server</td>
<td>Windows/Unix</td>
<td>3.1, 4.0, 4.1</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>GlassFish Server</td>
</tr>
<tr>
<td>Oracle HTTP Server</td>
<td>Windows/Unix</td>
<td>No data</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>HTTP Server</td>
</tr>
<tr>
<td>Oracle iPlanet Web Server</td>
<td>Solaris</td>
<td>4.x, 6.x, 7.x</td>
<td>SSH</td>
<td>Top-down and horizontal</td>
<td>Sun iPlanet Webserver</td>
</tr>
<tr>
<td>Oracle Java Enterprise System(JES)</td>
<td>Solaris</td>
<td>7.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>Sun JES pattern</td>
</tr>
<tr>
<td>Oracle Metric Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Metric Server</td>
</tr>
<tr>
<td>Oracle MySql Server</td>
<td>Windows/Unix</td>
<td>4.x, 5.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>My SQL server On Windows and Linux Pattern</td>
</tr>
<tr>
<td>Oracle Notification Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Oracle Notification Server</td>
</tr>
<tr>
<td>Oracle Peoplesoft</td>
<td>Windows/Unix</td>
<td>8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Peoplesoft Application Server</td>
</tr>
<tr>
<td>Oracle Process Manager</td>
<td>Windows/Unix</td>
<td>9.x.x, 10.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Process Manager</td>
</tr>
<tr>
<td>Oracle RAC</td>
<td>Windows/Unix</td>
<td>9.x.x, 10.x.x, 11.x.x, 12.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>App TNS Service</td>
</tr>
<tr>
<td>Oracle Report Server</td>
<td>Windows/Unix</td>
<td>9.x.x, 10.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Report Server</td>
</tr>
<tr>
<td>Oracle Solaris Server</td>
<td>Solaris</td>
<td>9, 10, 11, SPARC</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Solaris Server</td>
</tr>
<tr>
<td>Oracle Tuxedo</td>
<td>Windows/Unix</td>
<td>10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Tuxedo</td>
</tr>
<tr>
<td>Oracle Tuxedo Portal</td>
<td>Windows/Unix</td>
<td>10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Tuxedo Portal pattern</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------</td>
<td>--------------------------</td>
<td>---------------------</td>
<td>---------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>Oracle Weblogic Application Server</td>
<td>Windows/Unix</td>
<td>8.x, 9.x, 10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>WebLogic</td>
</tr>
<tr>
<td>Pivotal RabbitMQ</td>
<td>Windows/Unix</td>
<td>3.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>RabbitMQ Cluster Unix pattern</td>
</tr>
<tr>
<td>PostgreSQL DB</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>PostgreSQL DB</td>
</tr>
<tr>
<td>Puppet</td>
<td>Unix</td>
<td>N/A</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Probes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· Puppet – Master Info</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· Puppet – Certificate Requests</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· Puppet – Manifest</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· Puppet – Modules</td>
</tr>
<tr>
<td>RadWare AppDirector Load Balancer</td>
<td>RadWare</td>
<td>No data</td>
<td>SNMP</td>
<td>Top-down and horizontal</td>
<td>AppDirector Load Balancer</td>
</tr>
<tr>
<td>Red Hat Cluster Suite</td>
<td>Linux</td>
<td>RH 5.x, 6.x, 7.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>RH Cluster</td>
</tr>
<tr>
<td>SAP Convergent Invoicing (CI), including:</td>
<td>Windows, Linux, Hpxx, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP CI Application pattern</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· SAP Convergent Charging</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· SAP Customer Relationship Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>· Contract Accounts Receivable and Payable</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>-----------------------------------</td>
<td>---------</td>
<td>----------------</td>
<td>-------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>SAP NetWeaver Development Infrastructure (DI), including:</td>
<td>Windows, Linux, Hpux, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP DI Application pattern</td>
</tr>
<tr>
<td>- Change Management Service (CMS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Design Time Repository (DRT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Component Build Service (CBS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- System Landscape Directory (SLD and Name Service)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Common user management of the development infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Integration of the transports with the Change Transport System (CTS) of Application Server ABAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Available tools for monitoring the development infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAP Java Connector (JC)</td>
<td>Windows, Linux, Hpux, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP JC Application pattern</td>
</tr>
<tr>
<td>SAP Evaluated Receipt Settlement (ERS)</td>
<td>Windows, Linux, Hpux, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP ERS Application pattern</td>
</tr>
<tr>
<td>SAP Central Services (SCS)</td>
<td>Windows, Linux, Hpux, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP SCS Application pattern</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------</td>
<td>-------------------</td>
<td>---------------</td>
<td>-------------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>SAP Sybase DB</td>
<td>Windows/Unix</td>
<td>4.x, 5.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Sybase</td>
</tr>
<tr>
<td>Sun Directory Proxy Server</td>
<td>Solaris</td>
<td>No data</td>
<td>SSH</td>
<td>Top-down</td>
<td>Sun Directory Proxy Server</td>
</tr>
<tr>
<td>Sun Java System Directory Server</td>
<td>Solaris</td>
<td>No data</td>
<td>SSH</td>
<td>Top-down</td>
<td>Sun Directory Server</td>
</tr>
<tr>
<td>Thunder ADC</td>
<td>A10</td>
<td>2.x, 4.x</td>
<td>SNMP/SSH</td>
<td>Top-down and horizontal</td>
<td>A10 Load Balancer</td>
</tr>
<tr>
<td>TIBCO ActiveMatrix Adapter</td>
<td>Unix</td>
<td>7.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>Tibco Adapter</td>
</tr>
<tr>
<td>TIBCO ActiveMatrix BusinessWorks</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>ActiveMatrix Business Works</td>
</tr>
<tr>
<td>TIBCO Enterprise Message Service (EMS)</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Enterprise Message Service</td>
</tr>
<tr>
<td>TIBCO Enterprise Message Service (EMS) Queue</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Enterprise Message Service</td>
</tr>
<tr>
<td>TIBCO Hawk</td>
<td>Unix</td>
<td>4.x, 5.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>TibcoHawkPattern</td>
</tr>
<tr>
<td>Veritas Enterprise Vault</td>
<td>Windows</td>
<td>7.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Enterprise Vault</td>
</tr>
</tbody>
</table>

**Components installed with Service Mapping**

Several types of components are installed with activation of the Service Mapping plugin, including tables, user roles, properties, and scheduled jobs.

Demo data is available for this feature.
### Roles installed

<table>
<thead>
<tr>
<th>Role title (name)</th>
<th>Description</th>
<th>Contains roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>(app_service_admin)</td>
<td>Creates and modifies application services, creates service groups, views, and edits application service maps.</td>
<td>(itil)</td>
</tr>
<tr>
<td>(app_service_user)</td>
<td>Views application service maps. The itil role that serves as the basic helpdesk technician role contains the app_service_user role.</td>
<td>cmdb_read</td>
</tr>
<tr>
<td>(itom_admin)</td>
<td>Configure IT Operations Management applications using the ITOM Guided Setup.</td>
<td>(sn_bm_client.benchmark_data_viewer) (sm_itom_license.reader)</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The ITOM Guided Setup does not cover Service Mapping.</td>
<td></td>
</tr>
<tr>
<td>(sam_core_admin)</td>
<td>An internal role.</td>
<td>(sam_core_user)</td>
</tr>
<tr>
<td>(sam_core_user)</td>
<td>An internal role.</td>
<td>(itil)</td>
</tr>
<tr>
<td>(sm_admin)</td>
<td>SETS UP THE SERVICE MAPPING APPLICATION. Maps, fixes, and maintains application services. Also performs advanced configuration and customization of the product. Assign this role to application administrators.</td>
<td>• (app_service_admin) • (app_service_owner) • (discovery_admin) • (itil_admin) • (itom_admin) • (sam_core_admin) • (personalize_dictionary) • (pd_admin) • (pd_user)</td>
</tr>
<tr>
<td>(sm_app_owner)</td>
<td>PROVIDES INFORMATION NECESSARY FOR SUCCESSFUL MAPPING OF AN APPLICATION SERVICE. Once a service is mapped, this user reviews the results and either approves it or suggests changes. Assign the sm_app_owner role to users who own application services and are familiar with the infrastructure and applications that make up the services.</td>
<td>(sm_user)</td>
</tr>
<tr>
<td>(sm_itom_license.reader)</td>
<td>Views subscription information and statistics for IT Operations Management applications.</td>
<td></td>
</tr>
</tbody>
</table>
### Role title (name) | Description | Contains roles
--- | --- | ---
(sm_user) | Views maps for operational application services to plan change or migration, as well as analyze the continuity and availability of services. Assign this role to application users. | • (sam_core_user.sm_user)  
• (itil.sm_user)

(sn_svc_err.admin) | Fixes discovery errors in bulk using the error framework feature in Service Mapping and Discovery. | (itil)

(pd_admin) | Creates new and modifies existing patterns, which Service Mapping and Discovery use for performing horizontal and top-down discovery. | (pd_user)

### Scheduled jobs installed

<table>
<thead>
<tr>
<th>Scheduled job</th>
<th>Description</th>
</tr>
</thead>
</table>
| Deferred Discovery Cleanup  
(sysauto_script_ba4bee737f0022008f1c3b19be6efa910d) | Runs every minute. Detects and cancels discovery schedules on TCP connections that are not in the active state anymore (5 minutes of no activity). |
| Update Query Based Services  
(sysauto_script_08745913f3113102e6bae4716612b1f) | Runs every 10 minutes. Updates views for all defined technical services. |
| Update Business Service Status  
(sysauto_script_5641702ac32102003e7674e81d3ae26) | Runs every minute. Updates the discovery status attribute for application services. The discovery status is calculated as the overall status over all endpoints that an application service contains. |
| Flow Discovery Scheduler  
(sysauto_script_74c676f00d0b0220060ff742ea966291) | Runs every 10 minutes. Triggers the network flow discovery and writes the data into the (sa_flow_connector) table. |
| Save thread dumps  
(sysauto_script_a49c0d300c032020039a3553a81d3ae2) | Not active by default. When enabled, collects thread dumps and stores the data in the (sys_thread_dump) table. |
| Slow steps stats persist  
(sysauto_script_dc0d6e690c302020039a3553a81d3ae2) | Runs every minute. Collects data on slow execution of pattern steps and writes this data into the (sys_step_pattern) table. |
| Calculate number of servers  
(sys_trigger_db438f407f1502000128134f8d6a91c6) | Runs every Sunday. Updates the licensing metric with the overall number of discovered servers on instance. This information is used for billing. On domain separated instances, the number is updated for each domain. |
| Run DNS Lookup For All Entry Points and Manual Connection Endpoints  
(t178ab457fb213001952baf8be6fa91d3) | Runs the DNS lookup probe on all entry points and manual connection endpoints and updates information on their IP addresses in the CMDB. This scheduled job is disabled by default. |
### Scheduled Job

<table>
<thead>
<tr>
<th>Job Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(sm dedup tracker for specific discovery sources)</td>
</tr>
<tr>
<td>In some cases, Service Mapping and third-party</td>
</tr>
<tr>
<td>discovery or monitoring tools identify the same configuration item (CI)</td>
</tr>
<tr>
<td>differently. It results in the CMDB application table containing two records</td>
</tr>
<tr>
<td>for the same CI. This scheduled job scans the CMDB application tables that</td>
</tr>
<tr>
<td>contain information about CIs identified as generic applications. If the</td>
</tr>
<tr>
<td>scheduled job identifies duplicate records based on the CI type and the</td>
</tr>
<tr>
<td>discovery source, it creates a deduplication task.</td>
</tr>
<tr>
<td>(sys_auto_flush_521e7e1c320220039a3553a81d3ae56)</td>
</tr>
<tr>
<td>Cleans the (sa_flow_connection) table from records not updated for the last</td>
</tr>
<tr>
<td>seven days.</td>
</tr>
<tr>
<td>(sys_auto_flush_956d498b938503006a7031f6357fbee6)</td>
</tr>
<tr>
<td>Cleans the (sa_performance_statistics) table from records containing statistics of overall event processing time and created earlier than 30 days ago.</td>
</tr>
<tr>
<td>(sys_auto_flush_ce1a97819001200e0e931635f7fbd5b)</td>
</tr>
<tr>
<td>Cleans the (sa_performance_statistics) table from records created earlier than 2 days ago.</td>
</tr>
<tr>
<td>(sys_auto_flush_b24d9c30c302020039a3553a81d3ae56)</td>
</tr>
<tr>
<td>Cleans the (sys_thread_dump) table from records created earlier than 7 days ago.</td>
</tr>
<tr>
<td>(sys_auto_flush Cf129eb7f22320995cbaf8b61a919ed6)</td>
</tr>
<tr>
<td>Cleans the (sa_ml_feature_data) table from records not updated in the last 7 days, and have a special state (empty label field).</td>
</tr>
<tr>
<td>(sys_auto_flush_df16be1c320220039a3553a81d3ae56)</td>
</tr>
<tr>
<td>Cleans the (sa_flow_service) table from records not updated in the last 7 days.</td>
</tr>
<tr>
<td>(sys_auto_flush_dbf416be1c320220039a3553a81d3ae56)</td>
</tr>
<tr>
<td>Cleans the (sa_flow_server_comm) table from records not updated in the last 7 days.</td>
</tr>
</tbody>
</table>

### Tables installed

<table>
<thead>
<tr>
<th>Table Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre Service Mapping Task Scripts (sa_pre_task_script)</td>
</tr>
<tr>
<td>Contains scripts that Service Mapping runs on the instance and collect information that helps identify the relevant pattern for a CI.</td>
</tr>
<tr>
<td>Traffic Based Connection Qualifiers (cmdb_ci_qualifier_traffic_based_connection)</td>
</tr>
<tr>
<td>Legacy content. Not used in the current version.</td>
</tr>
<tr>
<td>Query Based Service (cmdb_ci_query_based_service)</td>
</tr>
<tr>
<td>Contains technical services.</td>
</tr>
<tr>
<td>Components (sa_bs_components)</td>
</tr>
<tr>
<td>Contains information on components that are part of the application service.</td>
</tr>
<tr>
<td>Customers enter this information manually in the Questionnaire tab of the</td>
</tr>
<tr>
<td>application service form.</td>
</tr>
<tr>
<td>Candidate Entry Points (sa_cand_entry_point)</td>
</tr>
<tr>
<td>Contains information on potential entry points that Service Mapping retrieved</td>
</tr>
<tr>
<td>from load balancers or traffic-based connections.</td>
</tr>
<tr>
<td>Dashboard View (sa_dashboard_view)</td>
</tr>
<tr>
<td>Contains configuration that defines which services to display in the Service tree.</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>Service Discovery Messages</td>
</tr>
<tr>
<td>ITOM Error task</td>
</tr>
<tr>
<td>Error EP AS Manual EP</td>
</tr>
<tr>
<td>Boundary endpoints</td>
</tr>
<tr>
<td>Manual connections</td>
</tr>
<tr>
<td>ML Feature Data</td>
</tr>
<tr>
<td>ML Model</td>
</tr>
<tr>
<td>ML Traffic Validation</td>
</tr>
<tr>
<td>Performance Statistics</td>
</tr>
<tr>
<td>Planned Custom Entry Point</td>
</tr>
<tr>
<td>Service Maps</td>
</tr>
<tr>
<td>Simulation entry points</td>
</tr>
<tr>
<td>Service Process Tasks</td>
</tr>
<tr>
<td>Slow Steps</td>
</tr>
<tr>
<td>Thread dump</td>
</tr>
<tr>
<td>Automation Error Category</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Automation Error Categories Statistics</td>
</tr>
<tr>
<td>(automation_error_category_stats)</td>
</tr>
<tr>
<td>Automation Error Code</td>
</tr>
<tr>
<td>(automation_error_code)</td>
</tr>
<tr>
<td>Automation Error Code Statistics</td>
</tr>
<tr>
<td>(automation_error_code_stats)</td>
</tr>
<tr>
<td>Automation Error Discovery Categories Statistics</td>
</tr>
<tr>
<td>(automation_error_discovery_category_stats)</td>
</tr>
<tr>
<td>(automation_error_discovery_code_stats)</td>
</tr>
<tr>
<td>Automation Error Discovery Message</td>
</tr>
<tr>
<td>(automation_error_discovery_msg)</td>
</tr>
<tr>
<td>Automation error instance category statistics</td>
</tr>
<tr>
<td>(automation_error_instance_category_stats)</td>
</tr>
<tr>
<td>Automation Error Categories Statistics</td>
</tr>
<tr>
<td>(automation_error_instance_code_stats)</td>
</tr>
<tr>
<td>Automation error instance message</td>
</tr>
<tr>
<td>(automation_error_instance_msg)</td>
</tr>
<tr>
<td>Automation error instance message update time</td>
</tr>
<tr>
<td>(automation_error_stats_update_time)</td>
</tr>
<tr>
<td>Automation Error Suggestion</td>
</tr>
<tr>
<td>(automation_error_suggestion)</td>
</tr>
<tr>
<td>Automation Source Application</td>
</tr>
<tr>
<td>(automation_source_application)</td>
</tr>
<tr>
<td>Mapping between suggestion and error code</td>
</tr>
<tr>
<td>(suggestion_error_code_m2m)</td>
</tr>
<tr>
<td>ITOM Error task</td>
</tr>
<tr>
<td>(sa_error_handler_task)</td>
</tr>
<tr>
<td>Applicative Credentials</td>
</tr>
<tr>
<td>(sa_applicative_credentials)</td>
</tr>
<tr>
<td>Service Mapping CI attributes</td>
</tr>
<tr>
<td>(sa_ci_attr)</td>
</tr>
<tr>
<td>Related CI Types</td>
</tr>
<tr>
<td>(sa_ci_to_pattern)</td>
</tr>
<tr>
<td>Pattern Debugger Session</td>
</tr>
<tr>
<td>(sa_debug_session)</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>Pattern Debugger Session Status for UI (sa_debug_session_status)</td>
</tr>
<tr>
<td>Sa Deferred Service Discovery (sa_deferred_service_discovery)</td>
</tr>
<tr>
<td>Service Discovery Log (sa_discovery_log)</td>
</tr>
<tr>
<td>Service Discovery Messages (sa_discovery_message)</td>
</tr>
<tr>
<td>Planned Business Service (sa_dw_business_service)</td>
</tr>
<tr>
<td>Components (sa_dw_components)</td>
</tr>
<tr>
<td>Check Connectivity Result (sa_dw_connectivity_result)</td>
</tr>
<tr>
<td>Connectivity Test (sa_dw_connectivity_test)</td>
</tr>
<tr>
<td>Planned Custom Entry Point (sa_dw_custom_entry_point)</td>
</tr>
<tr>
<td>Planned Entry Point (sa_dw_entry_point)</td>
</tr>
<tr>
<td>Planned Endpoint Attributes (sa_dw_ep_attrs)</td>
</tr>
<tr>
<td>Import Errors (sa_dw_import_errors)</td>
</tr>
<tr>
<td>Phase (sa_dw_phase)</td>
</tr>
<tr>
<td>Validation errors (sw_dw_validation_errors)</td>
</tr>
<tr>
<td>Sa Endpoint Status (sa_endpoint_status)</td>
</tr>
<tr>
<td>Find Host Scripts (sa_find_host_scripts)</td>
</tr>
<tr>
<td>nfdump file (sa_flcon_file_nfdump)</td>
</tr>
<tr>
<td>nfdump install (sa_flcon_local_nfdump)</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>AWS VPC flow logs</td>
</tr>
<tr>
<td>(sa_ficon_vpc_flow_log)</td>
</tr>
<tr>
<td>SA Flow Connection</td>
</tr>
<tr>
<td>(sa_flow_connection)</td>
</tr>
<tr>
<td>Flow Connector</td>
</tr>
<tr>
<td>(sa_flow_connector)</td>
</tr>
<tr>
<td>Flow Server Communication</td>
</tr>
<tr>
<td>(sa_flow_server_comm)</td>
</tr>
<tr>
<td>Flow Services IP/Port and Statistics</td>
</tr>
<tr>
<td>(sa_flow_service)</td>
</tr>
<tr>
<td>Host Class to Capability</td>
</tr>
<tr>
<td>(sa_host_class_to_capability)</td>
</tr>
<tr>
<td>Infra Path To Elements</td>
</tr>
<tr>
<td>(sa_infra_path_assoc)</td>
</tr>
<tr>
<td>Mapping External Discovery Commands</td>
</tr>
<tr>
<td>(sa_mapping_ext_commands)</td>
</tr>
<tr>
<td>Network Paths</td>
</tr>
<tr>
<td>(sa_network_paths)</td>
</tr>
<tr>
<td>Horizontal Discovered Paged Payloads</td>
</tr>
<tr>
<td>(sa_paged_payload)</td>
</tr>
<tr>
<td>Discovery Patterns</td>
</tr>
<tr>
<td>(sa_pattern)</td>
</tr>
<tr>
<td>Service Mapping Draft Pattern</td>
</tr>
<tr>
<td>(sa_pattern_draft)</td>
</tr>
<tr>
<td>Service Mapping Relation attributes</td>
</tr>
<tr>
<td>(sa_rel_attr)</td>
</tr>
<tr>
<td>Rel to Infra Path</td>
</tr>
<tr>
<td>(sa_rel_to_infra_path)</td>
</tr>
<tr>
<td>Service Mapping Server Usage</td>
</tr>
<tr>
<td>(sa_server_usage)</td>
</tr>
<tr>
<td>File System To Storage Path</td>
</tr>
<tr>
<td>(sa_storage_paths)</td>
</tr>
<tr>
<td>Storage Path Connectivity Map</td>
</tr>
<tr>
<td>(sa_storage_path_connectivity_map)</td>
</tr>
<tr>
<td>Traffic Based Discovery Rules</td>
</tr>
<tr>
<td>(sa_traffic_based_rules)</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>Uploaded File</td>
</tr>
<tr>
<td>Configuration Item</td>
</tr>
<tr>
<td>Application</td>
</tr>
<tr>
<td>IP Service</td>
</tr>
<tr>
<td>Application Category</td>
</tr>
<tr>
<td>Device Classification Category</td>
</tr>
<tr>
<td>Device Info Category</td>
</tr>
<tr>
<td>Protocol Category</td>
</tr>
<tr>
<td>HTTP Classification</td>
</tr>
<tr>
<td>HTTP Classification Match</td>
</tr>
<tr>
<td>Discovery Pattern Launcher</td>
</tr>
<tr>
<td>Cloud Execution Pattern</td>
</tr>
<tr>
<td>Serverless Execution Pattern</td>
</tr>
<tr>
<td>Discovery Pattern Parameters</td>
</tr>
<tr>
<td>Pattern Input Parameters</td>
</tr>
<tr>
<td>Applicative Credentials</td>
</tr>
<tr>
<td>Related CI Types</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>Custom operation</td>
</tr>
<tr>
<td>(sa_custom_operation)</td>
</tr>
<tr>
<td>Custom operation parameters</td>
</tr>
<tr>
<td>(sa_custom_operation_param)</td>
</tr>
<tr>
<td>Custom parsing strategy</td>
</tr>
<tr>
<td>(sa_custom_parsing_strategy)</td>
</tr>
<tr>
<td>Debug session</td>
</tr>
<tr>
<td>(sa_debug_session)</td>
</tr>
<tr>
<td>Pattern Debugger Session Status for UI</td>
</tr>
<tr>
<td>(sa_debug_session_status)</td>
</tr>
<tr>
<td>Sa Deferred Service Discovery</td>
</tr>
<tr>
<td>(sa_deferred_service_discovery)</td>
</tr>
<tr>
<td>Discovery log</td>
</tr>
<tr>
<td>(sa_discovery_log)</td>
</tr>
<tr>
<td>Find Host Scripts</td>
</tr>
<tr>
<td>(sa_find_host_scripts)</td>
</tr>
<tr>
<td>Host Class to Capability</td>
</tr>
<tr>
<td>(sa_host_class_to_capability)</td>
</tr>
<tr>
<td>Mapping Discovery Commands</td>
</tr>
<tr>
<td>(sa_mapping_ext_commands)</td>
</tr>
<tr>
<td>Horizontal Discovered Paged Payloads</td>
</tr>
<tr>
<td>(sa_paged_payload)</td>
</tr>
<tr>
<td>Discovery Patterns</td>
</tr>
<tr>
<td>(sa_pattern)</td>
</tr>
<tr>
<td>Service Mapping Draft Pattern</td>
</tr>
<tr>
<td>(sa_pattern_draft)</td>
</tr>
<tr>
<td>Pattern Extensions</td>
</tr>
<tr>
<td>(sa_pattern_extension)</td>
</tr>
<tr>
<td>Pattern Pre/Post Script</td>
</tr>
<tr>
<td>(sa_pattern_prepost_script)</td>
</tr>
<tr>
<td>Horizontal Discovered Payloads Snapshot</td>
</tr>
<tr>
<td>(sa_payload_snapshot)</td>
</tr>
<tr>
<td>Tracked File Definition</td>
</tr>
<tr>
<td>(sa_tracked_file_definition)</td>
</tr>
<tr>
<td>Table</td>
</tr>
<tr>
<td>-----------------------</td>
</tr>
<tr>
<td>Uploaded File</td>
</tr>
<tr>
<td>(sa_uploaded_file)</td>
</tr>
</tbody>
</table>

### Properties installed

**Note:** To open the System Property (sys_properties) table, enter `sys_properties.list` in the navigation filter.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>sa.storage_path_calculation.active</code></td>
<td>Enable storage path calculation.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Type:</strong> true/false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default value:</strong> true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Other possible values:</strong> false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Location:</strong> Service Mapping &gt; Administration &gt; Properties.</td>
</tr>
<tr>
<td><code>sa.update_paths_in_service_model.active</code></td>
<td>Update network and storage paths in Service Model.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Type:</strong> true/false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default value:</strong> true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Other possible values:</strong> false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Location:</strong> Service Mapping &gt; Administration &gt; Properties.</td>
</tr>
<tr>
<td><code>sa.network_path_calculation.active</code></td>
<td>Enable network path calculation.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Type:</strong> true/false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default value:</strong> true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Other possible values:</strong> false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Location:</strong> Service Mapping &gt; Administration &gt; Properties.</td>
</tr>
<tr>
<td><code>sa.discovery_task_timeout_min</code></td>
<td>Maximum time for a Service Mapping task in minutes (including waiting for execution in internal queues and ECC queue).</td>
</tr>
<tr>
<td></td>
<td>- <strong>Type:</strong> string</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default value:</strong> 120</td>
</tr>
<tr>
<td></td>
<td>- <strong>Other possible values:</strong> any number higher than 30</td>
</tr>
<tr>
<td></td>
<td>- <strong>Location:</strong> System Property (sys_properties) table</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| sa.debugger.max_timeout | Maximum timeout (in seconds) since the last server activity during a Pattern Debugger run.  
  - **Type**: integer  
  - **Default value**: 120  
  - **Other possible values**: any number higher than 60  
  - **Location**: Service Mapping > Administration > Properties. |
| sa.service_history.max_allowed_bubbles | Number of changes that the history scale and the Changes tab display.  
  - **Type**: integer  
  - **Default value**: 3000  
  - **Other possible values**: any number higher than 3000  
  - **Attention**: Increasing the value for this property may cause performance issues with loading application service maps.  
  - **Location**: Service Mapping > Administration > Properties. |
| sa.rediscovery.batch_size | Number of discovery tasks executed in a single batch.  
  - **Type**: integer  
  - **Default value**: 100  
  - **Other possible values**: any number higher than 10  
  - **Location**: Service Mapping > Administration > Properties. |
| sa.traffic_based_discovery.active | Traffic-based discovery  
  - **Type**: true/false  
  - **Default value**: true  
  - **Other possible values**: false  
  - **Location**: Service Mapping > Administration > Properties. |
| sa.traffic_based_discovery.conn_aging_time | Time period in hours for a Traffic Based Connection to remain active since last discovered.  
  - **Type**: integer  
  - **Default value**: 72  
  - **Other possible values**: any number higher than 24  
  - **Location**: Service Mapping > Administration > Properties. |
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| sa.traffic_based_discovery.ignored_ports | Ports to ignore when found by traffic-based discovery. This property is available in the System Property (sys_properties) table. Change this property to define ports that Service Mapping ignores while performing traffic-based discovery. It makes discovery more efficient since resources are not wasted on discovering irrelevant connections.  
  - **Type**: string  
  - **Default value**: 445, 139, 111, 2049, 860, 3260, 135, 53  
  - **Other possible values**: any relevant port numbers  
  - **Location**: System Property (sys_properties) table |
| sa.traffic_based_discovery.max_connections | Maximum number of traffic-based connections from a single CI. This property is available in the System Property (sys_properties) table. This property helps to keep the map size reasonable by limiting the number of possible CI connections.  
  - **Type**: integer  
  - **Default value**: 30  
  - **Other possible values**: any number higher than 1  
  - **Location**: System Property (sys_properties) table |
| best_practice.import_csv.max_file_size_in_kb | Maximal size of CSV files in KB. This property is used for importing application service candidates in bulk.  
  - **Type**: integer  
  - **Default value**: 5000  
  - **Other possible values**: any number higher than 1  
  - **Location**: Service Mapping > Service Planner > Properties. |
| glide.ui.sa_dw_business_service_activity.fields | Defines which fields are displayed for the activity feed in the Planner feature.  
  - **Type**: string  
  - **Default value**: *Attachments*, name, work_notes, *Email*, status, Business_service_owner  
  - **Other possible values**: comma-separated list of fields  
  - **Location**: System Property (sys_properties) table |
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sa.enable_gray_out_on_resume_discovery</td>
<td>Enable sub-tree to be grayed out after Service Mapping resumes discovery on an edge, or rerun discovery on the entire topology.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: boolean</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Other possible values</strong>: false</td>
</tr>
<tr>
<td></td>
<td>• <strong>Location</strong>: System Property (sys_properties) table</td>
</tr>
<tr>
<td>glide.service_mapping.computation_depth</td>
<td>This property controls the maximal depth of the application service map.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: integer</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: 25</td>
</tr>
<tr>
<td></td>
<td>• <strong>Other possible values</strong>: any number higher than 1</td>
</tr>
<tr>
<td></td>
<td>• <strong>Location</strong>: System Property (sys_properties) table</td>
</tr>
<tr>
<td>best_practice.import_csv.max_lines</td>
<td>Maximal number of lines in the CSV file used to import entry points.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: integer</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: 10000</td>
</tr>
<tr>
<td></td>
<td>• <strong>Other possible values</strong>: any number higher than 1</td>
</tr>
<tr>
<td></td>
<td>• <strong>Location</strong>: Service Mapping &gt; Service Planner &gt; Properties.</td>
</tr>
<tr>
<td>best_practice.check_connectivity.expand_listening</td>
<td>This property controls which PIDs Service Mapping checks during the hop operation in the connectivity check. This property is true, then Service Mapping only checks connectivity on PIDs whose ports are in the listening state, not connected.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: boolean</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Other possible values</strong>: false</td>
</tr>
<tr>
<td></td>
<td>• <strong>Location</strong>: System Property (sys_properties) table</td>
</tr>
<tr>
<td>sa.map.enable_loops_on_service_split</td>
<td>Enable you to connect a application service to another, already existing, application service even if the CI connection (endpoint) that you are connecting is part of this other application service. This property helps to avoid loops or cycles in service maps.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: boolean</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: false</td>
</tr>
<tr>
<td></td>
<td>• <strong>Other possible values</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Location</strong>: Service Mapping &gt; Administration &gt; Properties.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| `best_practice.check_connectivity.blacklist_ips` | Blacklist of IP addresses to expand to in the Connectivity Checks feature.  
- **Type**: string  
- **Default value**: 127.0.0.1, 127.0.0.2  
- **Other possible values**: comma-separated list of IP addresses  
- **Location**: Service Mapping > Service Planner > Properties. |
| `best_practice.check_connectivity.blacklist_ports` | Blacklist of ports to expand to in the connectivity checks feature.  
- **Type**: string  
- **Default value**: (empty)  
- **Other possible values**: comma-separated list of ports  
- **Location**: Service Mapping > Service Planner > Properties. |
| `best_practice.check_connectivity.max_ips` | The maximum number of IP addresses to process during the connectivity check.  
- **Type**: integer  
- **Default value**: 100  
- **Other possible values**: any number higher than 1  
- **Location**: Service Mapping > Service Planner > Properties. |
| `service_watch.implied_match.endpoint.black_list` | A list of endpoint types for which Service Mapping does not apply process matching logic utilized during the top-down pattern execution.  
- **Type**: string  
- **Default value**: `cmdb_ci_endpoint_ssis_mssql, cmdb_ci_endpoint_ssas_mssql`  
- **Other possible values**: comma-separated list of endpoints  
- **Location**: Service Mapping > Administration > Properties. |
| `sm.service.export.blacklist` | Exclude these fields from the record in the service definition export file.  
- **Type**: string  
- **Default value**: (empty)  
- **Other possible values**: comma-separated list of fields  
- **Location**: Service Mapping > Administration > Properties. |
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| sm.service.export.blacklist.cmdb_ci_endpoint | Excludes these fields from the entry point record in the service definition export file.  
- **Type**: string  
- **Default value**: (empty)  
- **Other possible values**: comma-separated list of fields  
- **Location**: Service Mapping > Administration > Properties. |
| sm.service.export.blacklist.cmdb_ci_service_discovered | Excludes these fields from the application service record in the service definition export file.  
- **Type**: string  
- **Default value**: discovery_status,operational_status,process_status,work_notes  
- **Other possible values**: comma-separated list of fields  
- **Location**: Service Mapping > Administration > Properties. |
| sm.service.export.blacklist.cmdb_ci_service_group | Excludes these fields from the service group record in the service definition export file. Use comma to separate fields.  
- **Type**: string  
- **Default value**: (empty)  
- **Other possible values**: comma-separated list of fields  
- **Location**: Service Mapping > Administration > Properties. |
| sm.service.export.max_services_limit | Defines the maximum allowed number of services to export.  
- **Type**: integer  
- **Default value**: 100  
- **Location**: Service Mapping > Administration > Properties. |
| sm.service.export.skip_empty_fields | Excludes fields with an empty value from the record in the service definitions export file.  
- **Type**: boolean  
- **Default value**: true  
- **Other possible values**: false  
- **Location**: Service Mapping > Administration > Properties. |
| sm.service.export.skip_reference_fields | Excludes reference fields from the record in the service definitions export file.  
- **Type**: boolean  
- **Default value**: true  
- **Other possible values**: false  
- **Location**: Service Mapping > Administration > Properties. |
### Service Mapping roles and flow

Learn about high-level tasks users having different roles perform in Service Mapping.

In Service Mapping, users with the following roles collaborate to map, review, and monitor organization application services:

**sm_admin**

Sets up the Service Mapping application. Maps, fixes, and maintains application services. Also performs advanced configuration and customization of the product. Assign this role to application administrators.

**sm_user**

Views maps for operational application services to plan change or migration, as well as analyze the continuity and availability of services. Assign this role to application users.

**sm_app_owner**

Provides information necessary for successful mapping of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. Assign the sm_app_owner role to users who own application services and are familiar with the infrastructure and applications that make up the services.

**Note:** For a complete list of Service Mapping roles, see Components installed with Service Mapping.

A typical Service Mapping workflow has the following stages:

1. The administrator performs basic obligatory configurations to set up Service Mapping.
2. The administrator maps organization business services in bulk. In addition, the administrator may map some application services individually.
3. The administrator fixes business service errors in bulk.
4. The administrator reviews the results of the initial mapping and fixes errors in individual business services.

---

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sm.service.export.skip_sys_fields</td>
<td>Excludes fields beginning with “sys_” from the record in the service definitions export file.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Type</strong>: boolean</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Other possible values</strong>: false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Location</strong>: Service Mapping &gt; Administration &gt; Properties.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sm.service.import.auto_start_discovery</td>
<td>Enable automatic discovery on all successfully imported entry points.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Type</strong>: boolean</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default value</strong>: false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Other possible values</strong>: true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Location</strong>: Service Mapping &gt; Administration &gt; Properties.</td>
</tr>
</tbody>
</table>
5. The administrator sends fixed business services to the business service owner for review.

6. The application service owner checks that the application service maps are complete and all major components comprising it are correctly represented. If necessary, the owner leaves comments, referred to as reject messages, on application service maps for the Service Mapping administrator to implement. See Review application service maps.

7. The administrator fine-tunes business service maps with feedback from the business service owner, and then resends them to the owner for review.

8. If the revised application service maps are satisfactory, the application service owner approves them. If not, the owner requests further fixes, which the administrator must address. See Review application service maps.

9. The administrator completes defining the business services by configuring access to them, as well as some advanced attributes like criticality.

10. After application service definition is complete, the Service Mapping user can view business service maps.
1. Set up Service Mapping
2. Map business services in bulk
3. Fix discovery errors in bulk
4. Fix discovery errors in individual maps
5. Send individual business services for review
6. Review business services and provide feedback
7. Implement owner’s requests and resend for review
8. Review and approve individual business services
9. Complete individual business service definition
10. View and analyze business service maps

In addition to these tasks, there may be a need to customize patterns that Service Mapping uses to discover devices and applications forming application services. Users must have the pd_admin role to customize patterns. In the base system, the sm_admin role contains the pd_role. Customizing patterns requires basic knowledge of programming.

Service Mapping setup

You get started with Service Mapping by configuring roles, credentials, and MID Server connections.

Setting up Service Mapping is the first stage in the Service Mapping workflow.
Service Mapping workflow

Service Mapping is part of the Now Platform and deploys some of its platform-wide mechanisms and features. At the same time, there are some configurations that are specific to Service Mapping only.

**Warning:** The Service Mapping setup for the Advance Release is different from the standard Service Mapping setup.
Perform the following tasks in the exact order they are listed below:

1. **Request Service Mapping.**

2. Install and configure MID Server. MID Servers, which are located in the enterprise private network, facilitate communication between servers on the network and some ServiceNow applications, such as Service Mapping, and Discovery. For more information, see [MID Server configuration for Service Mapping](#).

3. Verify that **Discovery** is set up and runs horizontal discovery as expected.

4. Configure **credentials required for host discovery.**

5. Configure credentials required for Service Mapping to access applications inside your organization private network. See [Prerequisites for performing top-down discovery using Service Mapping](#).

6. If your organization network contains load balancers running on a Linux host, create a **discovery behavior for load balancers.**

7. Grant the following Service Mapping roles to relevant users:

   - **sm_admin**
     
     Sets up the Service Mapping application. Maps, fixes, and maintains application services. Also performs advanced configuration and customization of the product. Assign this role to application administrators.

   - **sm_user**
     
     Views maps for operational application services to plan change or migration, as well as analyze the continuity and availability of services. Assign this role to application users.

   - **sm_app_owner**
     
     Provides information necessary for successful mapping of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. Assign the sm_app_owner role to users who own application services and are familiar with the infrastructure and applications that make up the services.

8. **Verify that Service Mapping is set up properly.**

9. If your organization has a ServiceNow deployment with customized Discovery or CMDB attributes, perform additional configuration described in [KB0647574: Preparing customized ServiceNow deployments to work with Service Mapping](#).

In addition to these obligatory setup configurations described here, you may need to perform additional configurations after you complete the initial mapping process. For more information, see [Advanced Service Mapping configuration](#).

---

**Request Service Mapping**

Service Mapping is available under the ITOM Visibility subscription and requires activation by ServiceNow personnel.

To purchase a subscription, contact your ServiceNow account manager. The account manager can arrange to have the plugin activated on your organization’s production and sub-production instances, generally within a few days.

If you do not have an account manager, decide to delay activation after purchase, or want to evaluate the product on a sub-production instance without charge, follow these steps.

Role required: admin
The following plugins are activated automatically when the Service Mapping plugin (com.snc.service-mapping) is activated:

- Discovery (com.snc.discovery)
- Pattern Designer (com.snc.pattern.designer)
- Cloud Management Core (com.snc.cloud.core)
- Performance Analytics – Content Pack – Service Mapping (com.snc.service-mapping.pa.content)
- Event Management and Service Mapping Core (com.snc.service-watch)

**Note:** The Event Management and Service Mapping Core (com.snc.service-watch) plugin is different from the Event Management plugin (com.glideapp.itom.snac).

1. From your instance, navigate to **System Definition > Plugins**.
2. On the All Applications page, click **Request Plugin** to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
Activate Plugin

In order to enhance the user experience, we have redesigned Activate a Plugin service catalog item. You can also use Manage Instances page on Service Portal to Activate a Plugin.

4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>Date and time must be at least two business days from the current time.</td>
</tr>
<tr>
<td>Reason/Comments</td>
<td>Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments.</td>
</tr>
</tbody>
</table>

5. Click Submit.

MID Server configuration for Service Mapping

Configure Service Mapping and MID Servers to work together.

What is the MID Server

The Management, Instrumentation, and Discovery (MID) Server is a Java application that runs as a Windows service or UNIX daemon on a server. MID Servers, which are located in the enterprise private network, facilitate communication between servers on the network and some ServiceNow applications, such as Service Mapping, and Discovery.

MID Server selection criteria

Most environments require multiple MID Servers, with Service Mapping using the relevant MID Server for discovery. MID Servers have the following mandatory selection criteria that Service Mapping uses to choose the relevant MID Server:

- Application — defines what application a MID Server works with. Set it to Service Mapping to reserve this MID Server exclusively to Service Mapping discovery requests. Alternatively, set it to ALL to allow any ServiceNow application to use this MID Server.
• Capability — defines the network capability. For Service Mapping, set this parameter to ALL or any combination of SSH, WMI, SNMP, and Cloud Management. See Configure MID Server capabilities.
• IP range — limits operation of this MID Server to this IP range. Service Mapping does not choose this MID Server for a discovery request whose endpoint is outside this IP range. See Configure an IP address range for the MID Server.

Important: These selection criteria are mandatory for the new MID Server algorithm. Configure them in every MID Server you are planning to use with Service Mapping.

Service Mapping selects a MID Server using the following algorithm:

• Service Mapping chooses the MID Server whose selection criteria best match the parameters of the discovery request.
• If there are no MID Servers with matching selection criteria, Service Mapping chooses the default MID Server.
• If there are no MID Servers with matching selection criteria or default MID Server, Service Mapping cannot start the discovery process.

While by default Service Mapping uses this algorithm in all deployments upgraded from Istanbul or Jakarta, it can support both new and legacy algorithms for selecting a MID Server. For more information, see Choose MID Server selection algorithm.

Default MID Servers for Service Mapping

In addition to selection criteria, you can configure one of the MID Servers as the default server that Service Mapping uses. If there are no MID Servers with matching application, capability, or IP range, Service Mapping uses the default MID Server. See Configure a default MID Server for each application.

Using PowerShell for discovery

MID Servers can use PowerShell to directly communicate with Windows servers using both WMI and WinRM protocols. For Windows services using the WinRM protocol, the PowerShell process establishes a secure PSSession (PowerShell Remoting session) that stays open until the MID Server finishes querying a Windows server. For Windows servers using the WMI protocol, the PowerShell process sends every PowerShell command with credentials.

If you do not configure MID Servers to use PowerShell and PowerShell Remoting, MID Servers use WMI Collector.

MID Server credential-less discovery with Nmap

If the MID Server does not have sufficient credentials to access a device or application, it can run Network Mapper (Nmap) commands to collect basic information without using credentials. Credential-less discovery with Nmap requires additional configuration as described in Install and uninstall Nmap on a MID Server.

ServiceNow applications refer to devices and applications that comprise an application service as configuration items (CIs).
Placing MID Servers

The number of MID Servers that you require and where you place them depends on your organization needs. If you want to map devices and applications inside your private network, place the MID Servers inside the private network. If you want to map devices and applications located in the DMZ, place the MID Servers both in the DMZ and inside the private network.

In deployments where domain separation is enabled and domains are configured to form a hierarchy, MID Servers must be placed in the lowest domain level, a "leaf domain".
Once MID Servers are installed, configure them to work with Service Mapping for the best discovery results.

*Choose MID Server selection algorithm*
Service Mapping supports the new and the legacy algorithms for selecting a MID Server for a discovery request. Depending on your organization needs, you can choose which algorithm to enable.

*Role required: admin or sm_admin*
Service Mapping selects a MID Server using the following algorithm:

- Service Mapping chooses the MID Server whose selection criteria best match the parameters of the discovery request.
- If there are no MID Servers with matching selection criteria, Service Mapping chooses the default MID Server.
If there are no MID Servers with matching selection criteria or default MID Server, Service Mapping cannot start the discovery process.

By default, Service Mapping uses this algorithm in all fresh installs and deployments upgraded from Istanbul or Jakarta.

In upgraded deployments, Service Mapping selects a MID Server using a legacy algorithm:

- Service Mapping chooses the MID Server whose application criteria is set to ServiceMapping or to ALL.
- By default, Service Mapping selects the MID Server whose IP range matches the IP in the discovery request.
- If Service Mapping cannot find a MID Server with the matching IP range and matching application, it assigns the discovery request to the default MID Server.
- If there is no default MID Server and none of the MID Servers have IP ranges configured for them, Service Mapping uses a random MID Server.
- If there is no default MID Server and no IP range matches the IP in the discovery request, Service Mapping cannot start the discovery process.

By default, Service Mapping uses the legacy algorithm in all deployments upgraded to Istanbul and earlier.

1. Navigate to Service Mapping > Administration > Properties.
2. To enable the new algorithm for deployments upgraded to Istanbul and earlier:
   a) Clear the Enable Legacy MID selection algorithm check box.
   b) Click Save.
   c) Configure MID Server selection criteria as described in MID Server configuration for Service Mapping.

3. To enable the legacy algorithm for the New York deployment:
   a) Select the Enable Legacy MID selection algorithm check box.
   b) Click Save.
   c) Configure MID Server selection criteria as described in MID Server configuration for Service Mapping in upgraded deployments.

MID Server configuration for Service Mapping in upgraded deployments

For ServiceNow deployments upgraded from earlier versions to Istanbul, Service Mapping uses the legacy algorithm to choose a MID Server for a discovery request.

MID Servers have the following selection criteria that Service Mapping in upgraded deployments uses to choose a MID Server for a discovery request:

- Application — defines what application a MID Server works with. Set it to Service Mapping to reserve this MID Server exclusively to Service Mapping discovery requests. Alternatively, set it to ALL to allow any ServiceNow application to use this MID Server.
  
  In earlier releases, you could configure MID Servers to work only with Service Mapping by using the Capability parameter. If you had MID Servers with this configuration, during the upgrade to Istanbul the application parameter is automatically set to ServiceMapping for them.

- IP range — limits operation of this MID Server to this IP range. Service Mapping does not choose this MID Server for a discovery request whose endpoint is outside this IP range.
In addition to the IP range, you can configure one of the MID Servers as the default server that Service Mapping uses. For operational information, see Configure a default MID Server for Service Mapping for upgraded deployments.

In upgraded deployments, Service Mapping selects a MID Server using a legacy algorithm:

- Service Mapping chooses the MID Server whose application criteria is set to **ServiceMapping** or to **ALL**.
- By default, Service Mapping selects the MID Server whose IP range matches the IP in the discovery request.
- If Service Mapping cannot find a MID Server with the matching IP range and matching application, it assigns the discovery request to the default MID Server.
- If there is no default MID Server and none of the MID Servers have IP ranges configured for them, Service Mapping uses a random MID Server.
- If there is no default MID Server and no IP range matches the IP in the discovery request, Service Mapping cannot start the discovery process.

The new algorithm for selecting MID Servers uses an extra selection criteria: capability. It allows Service Mapping to take a network protocol into consideration. For information about updated selection criteria and the new MID Server selection algorithm, see Choose MID Server selection algorithm.

While by default Service Mapping uses the legacy algorithm in upgraded deployments, it can support both new and legacy algorithms for selecting a MID Server. If necessary, you can enable the new MID Server selection algorithm for your upgraded deployment. For more information, see Choose MID Server selection algorithm.

Configure a default MID Server for Service Mapping for upgraded deployments

In deployments upgraded to Istanbul or earlier, Service Mapping uses the default MID Server when it cannot find a MID Server with the matching IP range. Configuring a default MID Server improves the discovery process.

Role required: sm_admin

Perform this procedure only for deployments upgraded to Istanbul or earlier. For fresh installs, perform Configure a default MID Server for each application.

Ensure that you know the name of the MID Server you want to configure as the default MID Server for Service Mapping.

In upgraded deployments, Service Mapping selects a MID Server using a legacy algorithm:

- Service Mapping chooses the MID Server whose application criteria is set to **ServiceMapping** or to **ALL**.
- By default, Service Mapping selects the MID Server whose IP range matches the IP in the discovery request.
- If Service Mapping cannot find a MID Server with the matching IP range and matching application, it assigns the discovery request to the default MID Server.
- If there is no default MID Server and none of the MID Servers have IP ranges configured for them, Service Mapping uses a random MID Server.
- If there is no default MID Server and no IP range matches the IP in the discovery request, Service Mapping cannot start the discovery process.

1. Navigate to System Properties by entering `sys_properties.list` in the navigation filter.
2. Click **New**.
3. Configure values for the new property as follows:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter mid.server.sm_default.</td>
</tr>
</tbody>
</table>
### Property | Description
--- | ---
Description | Enter a free text description of the use of this MID Server.
Type | Select **string**.
Value | Enter the name of the MID Server you want to configure as the default for Service Mapping.

4. **Click Update.**

*Create a dedicated WMI Collector service for MID Servers running on the same server*

A WMI (Windows Management Instrumentation) Collector service (wmi_collector.exe) is a MID Server component that helps it to communicate with Windows servers. If your deployment uses multiple MID Servers, make it more robust by creating a dedicated WMI Collector service for each MID Server.

1. Decide on the port number used by MID Server to communicate with the WMI collector.

2. Check that no other process uses this port.

   For example, to check if any process is already using the port 8585, use this netstat command:
   ```
   C:\temp>netstat -ano | findstr LISTEN | findstr 8585
   ```
   If the port is already in use, the output states it as follows:
   ```
   TCP 0.0.0.0:8585 0.0.0.0:0 LISTENING 4
   TCP [::]:8585 [::]:0 LISTENING 4
   ```

**Role required:** sm_admin

As default, MID Servers use a WMI (Windows Management Instrumentation) Collector service that helps MID Servers to communicate with Windows servers. Patterns used to discover Windows servers or applications running on them, contain WMI and WinRM queries and commands to run on Windows servers. A WMI Collector service transfers WMI and WinRM queries and commands from the MID Server to Windows-based CIs and brings the results of the queries to the MID Server.

MID Servers can use PowerShell to directly communicate with Windows servers using both WMI and WinRM protocols. For Windows services using the WinRM protocol, the PowerShell process establishes a secure PSSession (PowerShell Remoting session) that stays open until the MID Server finishes querying a Windows server. For Windows servers using the WMI protocol, the PowerShell process sends every PowerShell command with credentials.

If you do not configure MID Servers to use PowerShell and PowerShell Remoting, MID Servers use WMI Collector.

In some deployments, there is a default WMI Collector service shared by MID Servers installed on the same server, even if these MID Servers belong to different instances. This setup is potentially problematic, since this single WMI Collector service may be in non-operational state, be not upgraded, or experience overload issues.

**Note:** WMI Collector will be deprecated in a future release.
To make your deployment more robust, create a unique WMI Collector service for every MID Server running on the same server.
1. Access one of the instances.
2. Navigate to MID Servers > Servers.
3. Select the MID Server to which you want to add a WMI collector.
4. Create a WMI collector as follows:
   a) On the Configuration Parameters tab, click New.
   b) Define the collector attributes:
      | Field         | Description                                                                 |
      |---------------|-----------------------------------------------------------------------------|
      | Parameter name| Select mid.servicewatch.wmi_collector_service.                              |
      | Domain        | Do not change this setting. The instance retrieves it from the MID Server.  |
      | Value         | Enter the descriptive name for the new WMI Collector service.               |
   c) Click Submit.
5. Define the port for the WMI collector:
   a) On the Configuration Parameters tab, click New.
   b) Define the port attributes:
<pre><code>  | Field         | Description                                                               |
  |---------------|---------------------------------------------------------------------------|
  | Parameter name| Select mid.servicewatch.wmi.port.                                         |
</code></pre>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain</td>
<td>Do not change this setting. The instance retrieves it from the MID Server.</td>
</tr>
<tr>
<td>Value</td>
<td>Enter the port number on which the WMI collector receives data from the MID Server. By default, WMI collector uses port 8585. <strong>Important:</strong> Make sure that no other process uses this port.</td>
</tr>
</tbody>
</table>

3) Click **Submit**.

6. If necessary, repeat this procedure for other MID Servers running on the same server.

**Note:** The new service is not generated immediately. It is only generated upon running WMI steps in a Discovery schedule.

**Prerequisites for performing top-down discovery using Service Mapping**

Learn about credentials, users, and user permissions you must configure to let Service Mapping access and discover applications inside your organization private network.

Configure the relevant requirements to enable successful top-down discovery:

**Host credentials**

Service Mapping uses credentials of the hosts to access applications running on this host. Typically, host credentials are enough for top-down discovery. For example, to discover the HP Service Manager Application Server, configure only credentials for accessing the server hosting this application, SSH credentials, or Windows credentials.

**Applicative credentials**

To discover some applications, Service Mapping needs separate credentials from credentials of the host on which such applications run. This type of credentials is referred to in ServiceNow as applicative credentials. For example, to discover the ABAP SAP Central Services (ASCS), configure applicative credential in addition to the host credentials.

**Elevated rights for Unix-based hosts**

For Unix-based hosts, configure on MID Servers, you must provide a user with sudo-level credentials on all Unix-based hosts in your organization. For example, if the server hosting ASCS is Unix-based, provide a user with elevated rights to be able to run discovery commands against this Unix-based server.

**SNMP community credentials**

Configure SNMP community credentials to allow Service Mapping and Discovery to query network devices using the SNMP protocol. For example, configure SNMP community credentials for A10 Networks Thunder ADC.

**Users or user permissions**

You may need to provide additional users or user permissions to run certain commands or access certain directories. For example, for ASCS, provide a user with permissions to run SOAP on RFC read table function.
The Now Platform provides a centralized way of configuring credentials for Discovery, Service Mapping, and other applications. It allows you to configure credentials for hosts and applications only once. If you have already configured host-related credentials for another application, you do not need to do it again for Service Mapping.

You can assign credentials to specific MID Servers or keep credential generic so all MID Servers can use them. MID Servers retrieve commands and credentials from the ServiceNow instance and apply them to discover CIs inside organization private network.

Use the following guidelines to decide for which MID Server to create credentials:

- If all CIs, belonging to the same CI type, share a credential, you do not need to specify a MID Server for it. In that case, it is used on all MID Servers by default.

  **Note:** Specifying the MID Server when configuring credentials is optional.

- If CIs of the same type and on the same network have different credentials, configure these credentials and define order in which Service Mapping uses them when trying to connect to these CIs.

In addition to generic credentials, you configure on MID Servers, you must configure sudo-level credentials on all Unix-based hosts in your organization.

If the MID Server does not have sufficient credentials to access a device or application, it can run Network Mapper (Nmap) commands to collect basic information without using credentials. Credential-less discovery with Nmap requires additional configuration as described in Install and uninstall Nmap on a MID Server.

Review the list of credentials necessary to discover hosts and applications supported in the base system.

<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX OS</td>
<td>AIX Server (cmdb_ci_aix_server)</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| IBM z/OS         | IBM zOS server Server (cmdb_ci.ibm_zos_server) | - Run the USS service on the port 22 on the z/OS server.  
- On the Now Platform, configure SSH credentials.  
- Provide the user with the following permissions:  
  - To write into the /tmp directory.  
  - To see all user processes.  
  - To run REXX scripts. |
| HP-UX OS         | HPUX Server (cmdb_ci.hpux_server)          | Provide the user with the following permissions:  
- To write into the /tmp directory.  
- To see all user processes.  
- To run REXX scripts. |
| Linux OS         | Linux Server (cmdb_ci.linux_server)        | Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user. |
| Solaris OS       | Solaris Server (cmdb_ci.solaris_server)    | Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user. |
| Windows OS       | Windows Server (cmdb_ci.windows_server)    | On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.  
Provide all necessary credentials and permissions as described in Credentials required for host discovery. |
| A10 Load Balancing | A10 Load Balancer (cmdb_ci.lb_A10)        | Configure applicative credentials.                                                                 |

Configure SNMP credentials.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Active Directory Domain Services</td>
<td>Active Directory Domain Controller (cmdb_ci_ad_controller)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Amazon Web Services (AWS)</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td>Configure Amazon Web Services credentials.</td>
</tr>
<tr>
<td>Elastic Load Balancing</td>
<td></td>
<td>On the Now Platform, create a service account. Set the Account ID to the Amazon ELB account ID as it appears in the AWS Management Console.</td>
</tr>
<tr>
<td>Amazon Web Services (AWS) API Gateway</td>
<td></td>
<td>Configure Amazon Web Services credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, create a service account. Set the Account ID to the Amazon account ID to which API Gateway belongs. Use the Account ID as it appears in the AWS Management Console.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide a user with permissions to run the apigateway:GET function for just the GET action in API Gateway.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure to place the MID Server on the virtual machine that is part of the service account that you are planning to discover.</td>
</tr>
<tr>
<td>Amazon Web Services (AWS)</td>
<td>Windows Server (cmdb_ci_windows_server) or Linux Server (cmdb_ci_linux_server)</td>
<td>(If hosted on a Unix server), on the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>Elastic Compute Cloud (EC2)</td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, create a service account. Set the Account ID to the Amazon account ID to which API Gateway belongs. Use the Account ID as it appears in the AWS Management Console.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure to place the MID Server inside the Amazon cloud.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>--------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Amazon Web Services (AWS) Relational Database</td>
<td>Cloud DataBase</td>
<td>Configure Amazon Web Services credentials.</td>
</tr>
<tr>
<td>Service (RDS)</td>
<td>(cmdb_ci_cloud_database)</td>
<td>On the Now Platform, create a service account. Set the Account ID to the Amazon account ID to which RDS belongs. Use the Account ID as it appears in the AWS Management Console. Ensure to place the MID Server inside the Amazon cloud.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Amazon Web Services (AWS) Lambda</td>
<td>Cloud Function</td>
<td>Configure Amazon Web Services credentials.</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_cloud_function)</td>
<td>On the Now Platform, create a service account. Set the Account ID to the Amazon account ID to which AWS Lambda belongs. Use the Account ID as it appears in the AWS Management Console. Ensure to place the MID Server on the virtual machine that is part of the service account that you are planning to discover.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apache Tomcat</td>
<td>Tomcat</td>
<td>Give the user elevated rights to execute Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_app_server_tomcat)</td>
<td>(If hosted on a Unix server), on the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Apache Tomcat WAR</td>
<td>Tomcat WAR</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>(If hosted on Unix), on the Now Platform, configure <strong>SSH credentials</strong>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(If hosted on a Windows server)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>(Optional)</strong> For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td></td>
<td>(If hosted on Unix)</td>
<td>Give the user elevated rights to run <strong>Service Mapping commands requiring a privileged user</strong>.</td>
</tr>
<tr>
<td></td>
<td>(If hosted on a Windows server)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>(Optional)</strong> For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td></td>
<td>(If hosted on Unix), on the Now Platform, configure <strong>SSH credentials</strong>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(If hosted on a Windows server)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>(Optional)</strong> For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>**Give the user elevated rights to run <strong>Service Mapping commands requiring a privileged user</strong>.</td>
</tr>
<tr>
<td></td>
<td>(If hosted on a Windows server)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>(Optional)</strong> For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td></td>
<td>(If hosted on Linux or Unix), on the Now Platform, configure <strong>SSH credentials</strong>.</td>
<td>**Give the user elevated rights to run <strong>Service Mapping commands requiring a privileged user</strong>.</td>
</tr>
<tr>
<td></td>
<td>(If hosted on a Windows server)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>(Optional)</strong> For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>----------------</td>
</tr>
</tbody>
</table>
| BIG-IP Global Traffic Manager (GTM) F5 | Load Balancer Service (cmdb_ci_lb_service) | - Configure **SNMP credentials**.  
  
  **Note:** If you do not want to use SSH credentials, you can use the REST API to create a connection to F5 BIG-IP devices.  
  
  - (Optional) If there are iRules or SNMP community credentials are not enough for discovering outgoing connections, configure **SSH credentials** on Now Platform. Service Mapping uses the SSH credentials to retrieve connections not from the CMDB. Discovering connections using the SSH protocol is a failover mechanism for the SNMP-based discovery.  
  
  - Use **basic authentication** credentials for discovery via REST.  
  
  - (Optional) For top-down discovery with Service Mapping, configure **applicative credentials**.  
  
  - (Optional) For top-down discovery with Service Mapping, provide a user with permissions to run the following commands:  
    - bigpipe commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 9)  
    - bigpipe and Traffic Management Shell (TMSH) commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 10)  
    - Traffic Management Shell (TMSH) commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 11)  
    - Traffic Management Shell (TMSH) advanced commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 10, 11, and 12)  
    - The `show cm traffic-group - get` command for discovering F5 BIG-IP Device Service Clustering  
  
  - (Optional) For top-down discovery with Service Mapping, give the user **elevated rights to be able to run Service Mapping commands requiring a privileged user**.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIG-IP Local Traffic Manager (LTM) F5</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user. Provide a user with permissions to run the following commands:  - bigpipe commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 9)  - bigpipe and Traffic Management Shell (TMSH) commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 10)  - Traffic Management Shell (TMSH) commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 11)  - Traffic Management Shell (TMSH) advanced commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 10, 11, and 12)  Configure applicative credentials.  Configure SNMP community credentials as described in SNMP community credentials. (Optional) If there are iRules or SNMP community credentials are not enough for discovering outgoing connections, configure SSH credentials on Now Platform. Service Mapping uses the SSH credentials to retrieve connections not from the CMDB. Discovering connections using the SSH protocol is a failover mechanism for the SNMP-based discovery. (Optional) Configure Basic authentication credentials.</td>
</tr>
<tr>
<td>BMC CTRL-M Enterprise Manager</td>
<td>Control-M (cmdb_ci_appl_controlm)</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user. (If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BMC CTRL-M Gateway</td>
<td>Control-M (cmdb_ci_appl_controlm)</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Linux or Unix server) (Optional) on the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Linux or Unix server) (Optional) on the Now Platform, configure cc.</td>
</tr>
<tr>
<td>BMC IT Asset Management (ITAM)</td>
<td>ITAM Asset Center (cmdb_ci_appl_itam)</td>
<td>On the Now Platform, configure a Windows OS user with permissions to access the ITAM registry as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td>CA Identity Manager Provisioning Server</td>
<td>CA Identity Manager Provisioning Server</td>
<td>On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_appl_ca_id_man)</td>
<td>(Optional) For discovering using the WinRM protocol, see <strong>Configure WinRM trusted hosts on MID Server</strong>.</td>
</tr>
<tr>
<td>Cadebill Interconnect</td>
<td>inter connect (cmdb_ci_inter_connect)</td>
<td>On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td>Cadebill Interconnect Instance</td>
<td>Interconnect Instance (cmdb_ci_interconnect_instance)</td>
<td>On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td>Cisco Application Control Engine ACE (on Cisco CSM)</td>
<td>ACE (cmdb_ci_lb_ace)</td>
<td>On the Now Platform, configure <strong>applicative credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, configure SNMP community credentials as described in <strong>SNMP community credentials</strong>.</td>
</tr>
<tr>
<td>Cisco CallManager</td>
<td>Cisco CallManager (cmdb_ci_appl_cisco_call_man)</td>
<td>(If hosted on a Linux or Unix server) (Optional) on the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td>Cisco Global Site Selector (GSS)</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td>Configure SNMP community credentials as described in <strong>SNMP community credentials</strong>.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide a user who has permissions to run query on the domain list, DNS rule output, and VIP answer to get the outgoing connections of GSS.</td>
</tr>
<tr>
<td>Citrix Delivery Controller</td>
<td>Delivery Controller (cmdb_ci_appl_delivery_controller)</td>
<td>• Configure <a href="#">Windows administrative credentials</a> to access the host machine.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Make sure that <a href="#">PowerShell for Discovery</a> is installed on your MID Servers. The patterns used for Citrix discovery require the ability to run PowerShell commands.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (Optional) For top-down discovery with Service Mapping, on Citrix Delivery Controller, install the Citrix.Broker module.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (Optional) For top-down discovery with Service Mapping, a user with permission to run PowerShell commands.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (Optional) (Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Citrix Usage Collector   | Citrix Collector (cmdb_ci_appl_citrix_collector) | • Configure applicative credentials.  
• Configure a user with permissions to run PowerShell commands on the Citrix server.  
• (If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.  

**Note:** To use this credential later for XenAPP, set the Credential alias attribute to cmdb_ci_appl_citrix_xenapp. To use this credential later for Citrix, set the Credential alias attribute to cmdb_ci_appl_citrix_collector.  

• To discover Epic, provide permission to run a farm command.  
• Provide a user with permissions to read and query the Citrix repository:  
  • For versions 4.5 and later, a PowerShell script from the XenApp server.  
  • For versions earlier than 4.5, a Microsoft Visual Basic Script (VBScript) from the Citrix Presentation Server  
• (If hosted on a Unix server) On the Now Platform, configure SSH credentials.  

**Note:** To use this credential later for XenAPP, set the Credential alias attribute to cmdb_ci_appl_citrix_xenapp. To use this credential later for Citrix, set the Credential alias attribute to cmdb_ci_appl_citrix_collector.  

<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
</table>
| Citrix Presentation Server/Citrix XenApp             | Citrix XenAPP or Presentation Server (cmdb_ci_appl_citrix_xenapp) | • On the Now Platform, configure a domain user for accessing the Windows OS as described in [Windows credentials](#).  
  • (Optional) To discover using the WinRM protocol, see [Configure WinRM trusted hosts on MID Server](#). |
|                                                      |                                                           | Note: Citrix XenApp is the new name for Presentation Server beginning with the XenApp 5 release.                                             |
| Citrix Netscaler                                     | Citrix Netscaler (cmdb_ci_lb_netscaler)                   | • Configure [SNMP credentials](#).  
  • (Optional) Provide a user with the read access to the NS.conf configuration file if you do not want to provide SNMP or SSH credentials.  
  • Use [basic authentication](#) credentials for discovery via REST.  

Service Mapping uses the SSH credentials to retrieve connections not from the CMDB. Discovering connections using the SSH protocol is a failover mechanism for the SNMP-based discovery.  

Note: You also have the option of using [SSH credentials](#). However, if the payload size of the data that is returned to the instance is large, the discovery process can fail. In this case, use SNMP credentials instead.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citrix Netscaler Global Server</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td>• Configure <a href="#">SNMP credentials</a>.</td>
</tr>
<tr>
<td>Load Balancer</td>
<td></td>
<td><strong>Note:</strong> You also have the option of using <a href="#">SSH credentials</a>. However, if the payload size of the data that is returned to the instance is large, the discovery process can fail. In this case, use SNMP credentials instead.</td>
</tr>
<tr>
<td>Connect APK</td>
<td>Connect-It Service (cmdb_ci_appl_connectit)</td>
<td>• (Optional) Provide a user with the read access to the NS.conf configuration file if you do not want to provide SNMP or SSH credentials.</td>
</tr>
<tr>
<td><em>(Optional)</em></td>
<td></td>
<td><strong>Use basic authentication credentials</strong> for discovery via REST.</td>
</tr>
<tr>
<td><em>(Optional)</em></td>
<td></td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td><em>(If hosted on a Linux or Unix server)</em></td>
<td><em>(Optional)</em> on the Now Platform, configure SSH credentials.</td>
<td></td>
</tr>
<tr>
<td><strong>EMC Documentum</strong></td>
<td>Documentum DocBase (cmdb_ci_appl_doc_docbroker)</td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td><em>(If hosted on a Linux or Unix server)</em></td>
<td><em>(Optional)</em> on the Now Platform, configure SSH credentials.</td>
<td></td>
</tr>
<tr>
<td><strong>Epic</strong></td>
<td>epicd app server (cmdb_ci_epicd_app_server)</td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>epic agent (cmdb_ci_epic_agent)</td>
<td></td>
<td><em>(If hosted on a Linux or Unix server)</em> (Optional) on the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>epic cache (cmdb_ci_epic_cache)</td>
<td></td>
<td><em>(If hosted on a Linux or Unix server)</em> (Optional) on the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>epicd (cmdb_ci_endpoint_epicd)</td>
<td></td>
<td><em>(If hosted on a Linux or Unix server)</em> (Optional) on the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FormEngine</td>
<td>Oracle Forms Engine (cmdb_ci_appl_ora_forms)</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Unix server)</em> On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>HAProx</td>
<td>HAProx Load Balancer (cmdb_ci_lb_haproxy)</td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>HP Operations Manager</td>
<td>HP Operations Manager (cmdb_ci_appl_hp_operations)</td>
<td><em>(If hosted on a Unix server)</em> On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>HP Service Manager Application Server</td>
<td>HP Service Manager (cmdb_ci_appl_hp_service)</td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>HP Service Manager Index</td>
<td>HP SM Index Server (cmdb_ci_appl_hp_index)</td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>HP Service Manager Knowledge Base</td>
<td>HP SM KnowledgeBase (cmdb_ci_appl_hp_sm_kb)</td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Unix server)</em> On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>HP Quality Center</td>
<td>HP Quality Center (cmdb_ci_appl_hp_qc)</td>
<td>On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>HP uCMDB</td>
<td>HP uCMDB (cmdb_ci_app_server_hp_ucmdb)</td>
<td>On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| IBM WebSphere Message Broker       | IBM WebSphere Message Broker (cmdb_ci_appl_ibm_wmb) | (If hosted on a Windows server) On the Now Platform, configure **Windows credentials**.  
(If hosted on Unix server) On the Now Platform, configure **SSH credentials** for a user with permissions to run the following commands:  
- ‘source " + $directory + "/mqsiprofile”  
- ‘echo $MQSI_VERSION ”  
- $directory + "/mqssreportbroker " + $name  
- $directory + "/mqssibrowse ‘ + $name + " -t BROKERRESOURCES “  
Set the **Credential alias** attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns. |
| IBM CTRL-M Server                  | Control-M (cmdb_ci_appl_contolm)                  | (If hosted on Unix) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.  
(If hosted on Windows), on the Now Platform, configure Windows credentials. |
| IBM CICS Transaction Gateway CTG   | IBM CTG (cmdb_ci_appl_ibm_ctg)                    | Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.  
(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. |
| IBM Customer Information Control System (CICS) | IBM CICS (cmdb_ci_appl_ibm_cics)                   | (If hosted on Unix) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.  
(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. |
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM DB2</td>
<td>DB2 Instance (cmdb_ci_db_db2_instance)</td>
<td>• (If hosted on Linux) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (If hosted on Windows), on the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Set the tag attribute to (cmdb_ci_db_db2_instance). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) To discover connections to IBM DB2, configure SSH credentials for a user with permissions to run the following commands:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• db2 list tables spaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• `db2cmd /c /w /i db2 -tvf +$ script_name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (If hosted on z/OS) Provide the user with the following permissions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To write into the /tmp directory.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To see all user processes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To run REXX scripts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To read into the db2 folder.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (If hosted on z/OS) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>IBM J2EE EAR</td>
<td>(cmdb_ci_app_server_ws_ear)</td>
<td>(If hosted on Linux only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Windows), on the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>----------------</td>
</tr>
</tbody>
</table>
| IBM Tivoli Access Manager WebSEAL or IBM Security Access Manager appliance | Webseal (cmdb_ci_app_server_webseal) | (For IBM Security Access Manager appliance only) Configure **applicative credentials** for the user with permissions to run the following REST commands:  
  - `/reverseproxy` – for retrieving data about the reverse proxy and instances.  
  - `/wga/reverseproxy/*+$junctionstable().instance*/junctions` – for getting junctions per each instance.  
  - `/wga/reverseproxy/*+$instance_name*/junctions?junctions_id=$junction_name` – for getting forwarding URLs for the entry point junction.  
  (If hosted on Linux) On the Now Platform, configure **SSH credentials**.  
  (If hosted on Windows), on the Now Platform, configure **Windows credentials**.  
  (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. |
| IBM WebSphere Application Server | IBM Websphere (cmdb_ci_app_server_websphere) | (If hosted on Unix only) Give the user elevated rights to be able to run **Service Mapping commands requiring a privileged user**.  
  (If hosted on Linux) On the Now Platform, configure **SSH credentials**.  
  (If hosted on Windows), on the Now Platform, configure **Windows credentials**.  
  (If hosted on z/OS) On the Now Platform, configure **SSH credentials**. |
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(If hosted on z/OS) Provide the user with the following permissions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- To write into the /tmp directory.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- To see all user processes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- To run REXX scripts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- To run commands using oeconsol.</td>
</tr>
</tbody>
</table>
|                                   |                                              | - To run the command that brings info on the WebSphere Application Server task: `oeconsol 'D OMVS, A=ALL' | grep "+$taskname`.
<p>|                                   |                                              | - To read content of the WAS installation folder and the permission to read was.env.file where the installation variables are situated.        |
| (Optional) For discovering using the WinRM protocol, see |                                              | Configure WinRM trusted hosts on MID Server.                                                                                               |
| IBM WebSphere DataPower          | Data Power (cmdb_ci_app_server_datapower)    | Configure applicable credentials as described in SNMP community credentials.                                                                                                                            |
|                                   |                                              | Configure SNMP community credentials as described in SNMP community credentials.                                                            |
|                                   |                                              | (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.                                                                                               |
| IBM WebSphere Message Broker Flow | WMB Flow (cmdb_ci_appl_wmb)                   | (If hosted on a Unix server) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.                                                                    |
|                                   |                                              | Provide an IBM WebSphere Message Broker OS user with permissions to run the WebSphere Message Broker service.                                                                                           |
|                                   |                                              | (If hosted on Windows), on the Now Platform, configure Windows credentials.                                                                  |</p>
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>(If hosted on a Unix server) On the Now Platform, configure <strong>SSH credentials</strong> for a user with permissions to run the following commands:</td>
<td>(If hosted on a Unix server) On the Now Platform, configure <strong>SSH credentials</strong> for a user with permissions to run the following commands:</td>
<td>Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td>· “source ” + $directory + “/ mqsiprofile”</td>
<td>· “source ” + $directory + “/ mqsiprofile”</td>
<td>Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td>· 'echo $MQSI_VERSION ” $directory + ‘/ mqsireportbroker ” + $name</td>
<td>· 'echo $MQSI_VERSION ” $directory + ‘/ mqsireportbroker ” + $name</td>
<td>Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td>· $directory + ’/ mqsibrowse ” + $name + ’ -t BROKERRESOURCES “</td>
<td>· $directory + ’/ mqsibrowse ” + $name + ’ -t BROKERRESOURCES “</td>
<td>Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td>IBM WebSphere Message Broker</td>
<td>IBM WebSphere Message Broker (cmdb_ci_appl_ibm_wmb)</td>
<td>(If hosted on Unix) Give the user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user.</strong></td>
</tr>
<tr>
<td>(If hosted on a Unix server) On the Now Platform, configure <strong>SSH credentials</strong>. Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
<td>(If hosted on a Unix server) On the Now Platform, configure <strong>SSH credentials</strong>. Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_ibm_wmb). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
<td>(If hosted on a Windows server) On the Now Platform, configure <strong>Windows credentials.</strong></td>
</tr>
<tr>
<td>IBM WebSphere MQ</td>
<td>IBM WebSphere MQ (cmdb_ci_appl_ibm_wmq)</td>
<td>(If hosted on Unix) Give the user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user.</strong></td>
</tr>
<tr>
<td>(If hosted on Unix) Give the user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user.</strong></td>
<td>(If hosted on Unix) Give the user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user.</strong></td>
<td>(If hosted on Unix) Give the user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user.</strong></td>
</tr>
</tbody>
</table>
Configure applicative credentials for a user with permissions to run the following commands:

- (If hosted on a Windows server) `echo display QMSTATUS ALL | runmqsc -e + $queue_manager` for getting the queue manager (QM) status.
- (If hosted on a Windows server) `echo display QMGR ALL | runmqsc -e + $queue_manager` for getting the complete information about QM.
- (If hosted on a Unix server) `echo " display QMSTATUS ALL " | runmqsc " + $queue_manager` for getting the QM status.
- (If hosted on a Unix server) `echo " display QMGR ALL " | runmqsc " + $queue_manager` for getting the complete information about QM.
- (If hosted on a Unix server) `echo " display queue(" + $entry_point.queue + ") " | runmqsc " + $queue_manager` for getting the information about the EP queue.

(If hosted on Windows), on the Now Platform, configure Windows credentials.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(If hosted on a Unix server) configure <strong>SSH credentials</strong>. While configuring OS credentials, set the <strong>Credential alias</strong> attribute to <code>cmdb_ci_appl_ibm_wmq</code>. Configuring this attribute allows you later to use the credential for this CI in patterns. (Optional) To find IBM WebSphere MQ connections, configure SSH credentials for a user with permissions to run the following commands:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• `echo &quot;display queue&quot;(+$entry_point.queue++) &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• `echo &quot;display QMSTATUS ALL &quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• `echo &quot;display QMGR ALL &quot;</td>
</tr>
<tr>
<td>IBM WebSphere MQ Queue</td>
<td>IBM WebSphere MQ Queue (cmdb_ci_appl_ibm_wmq_queue)</td>
<td>(If hosted on a Unix server) Give the user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user</strong>.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
</table>

Configure **applicative credentials** for a user with permissions to run the following commands:

- `dspmq` – for getting information about the queue manager.

- `"echo \" display queue(" + $entry_point.queue + ") \" I runmqsc " + $queue_manager` – for getting the queue information.

- `"echo \" display clusqmgr(*) conname \" I runmqsc " + $queue_manager` – for getting all cluster queue manager (QM) lines.

- `"echo \" display qstatus(" + $entry_point.queue + ") TYPE(HANDLE) OPENTYPE(INPUT) CONNAME PID \" I runmqsc " + $queue_manager` – for getting the queue status information.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>(If hosted on Unix server) on the Now Platform, configure <strong>SSH credentials</strong> for a user with permissions to run the following commands:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· dspmq</td>
<td></td>
<td></td>
</tr>
<tr>
<td>echo &quot; display queue(&quot;+$entry_point.queue+&quot;) &quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>runmqsc &quot;+$queue_manager&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· (Optional) &quot;echo &quot; display qcluster(&quot;+$rqmname+&quot;) &quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>runmqsc &quot;$queue_manager&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· (Optional) &quot;echo &quot; dischl(*) CHLTYPE(SDR) CONNAME XMITQ WHERE(XMITQ EQ &quot;+$xmit_name+&quot;)&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>runmqsc &quot;$queue_manager&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· (Optional) &quot;echo &quot; display clusqmgr(*) conname &quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>runmqsc &quot;$queue_manager&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· (Optional) &quot;echo &quot; display qstatus(&quot;+$entry_point.queue +&quot;) TYPE(HANDLE) OPENTYPE(INPUT) CONNAME PID &quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>runmqsc &quot;$queue_manager&quot;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

While configuring OS credentials, set the **Credential alias** attribute to `cmdb_ci_appl_ibm_wmq`. Configuring this attribute allows you later to use the credential for this CI in patterns.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(If hosted on Windows), on the Now Platform, configure Windows credentials for a user with permissions to run the following commands:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- dspmq</td>
</tr>
<tr>
<td></td>
<td></td>
<td>echo &quot; display queue(&quot;+ $entry_point.queue+&quot;) &quot; I runmqsc +$queue_manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- (Optional) &quot;echo &quot; display aclustert(&quot;+$arnname+&quot;) &quot; I runmqsc +$queue_manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- (Optional) &quot;echo &quot; displsch(*) CHLTYPE(SDR) CONNAME XMITQ WHERE(XMITQ EQ '+ $xmit_name+')&quot; I runmqsc +$queue_manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- (Optional) &quot;echo &quot; display clusqmgr(*) conname &quot; I runmqsc +$queue_manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- (Optional) &quot;echo &quot; display qstatus(&quot;+$entry_point.queue +&quot;) TYPE(HANDLE) OPENTYPE(INPUT) CONNAME PID &quot; I runmqsc +$queue_manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While configuring OS credentials, set the Credential alias attribute to cmdb_ci_appl_ibm_wmq. Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>IBM WebSphere Portal</td>
<td>Websphere Portal</td>
<td>(If hosted on Linux only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_appl_webosphere_portal)</td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Interconnect Web Service</td>
<td>Inter connect (cmdb_ci_inter_connect)</td>
<td>(If hosted on a Unix server) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>JBoss</td>
<td>JBoss (cmdb_ci_app_server_jboss)</td>
<td>Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix server) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Jboss Module</td>
<td>Jboss module (cmdb_ci_app_server_jb_module)</td>
<td>(If hosted on a Unix server) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>JRun</td>
<td>Jrun (cmdb_ci_app_server_jrun)</td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>JRun WAR Inc</td>
<td>Jrun WAR (cmdb_ci_app_server_jrun_war)</td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>----------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Microsoft Azure Web Sites</td>
<td>Cloud WebServer (cmdb_ci_cloud_webserver)</td>
<td>On the Now Platform, add an Azure service principal and set up a cloud account for Azure. Set the account Id to the Azure Service Principal as it appears at the Microsoft Azure Console.</td>
</tr>
<tr>
<td>Microsoft Azure Load Balancer</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td>On the Now Platform, configure add an Azure service principal and set up a cloud account for Azure. Set the account Id to the Azure Service Principal as it appears at the Microsoft Azure Console.</td>
</tr>
<tr>
<td>Microsoft Azure SQL Database</td>
<td>Cloud Database Rule (cmdb_ci_cloud_database)</td>
<td>On the Now Platform, add an Azure service principal and set up a cloud account for Azure. Set the account Id to the Azure Service Principal as it appears at the Microsoft Azure Console.</td>
</tr>
<tr>
<td>Microsoft Dynamic CRM</td>
<td>Dynamic CRM Component (cmdb_ci_appl_ms_dynamic_crm)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials. (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Microsoft BizTalk Orchestration</td>
<td>BizTalk Orchestration (cmdb_ci_appl_biztalk_orch)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials. (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Microsoft BizTalk Server</td>
<td>BizTalk (cmdb_ci_appl_biztalk)</td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_exchange_frontend)</td>
<td>Configure credentials required for host discovery for a Windows server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Microsoft FAST Search Server</td>
<td>Fast Search</td>
<td>Create a user with the permission to read the following files:</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_appl_fastsearch)</td>
<td>• install_directory +&quot;\etc\VERSION.xml</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• and install_directory +&quot;\etc\CSConfig.xml</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Microsoft Exchange Hub</td>
<td>ExchangeHub (cmdb_ci_exchange_hub)</td>
<td>• On the Now Platform, configure a domain user for accessing the Windows OS as described in <a href="#">Windows credentials</a>. Use the Microsoft Exchange Server administrator’s user name and password as Windows credentials. For example, user name: local \ administrator and password: Are1300.</td>
</tr>
<tr>
<td>Transport Server</td>
<td>Exchange MailBox (cmdb_ci_exchange_mailbox)</td>
<td>Note: If you configure Windows credentials for a user different from the Microsoft Exchange Server administrator, Service Mapping uses traffic-based discovery that results in mapping all Microsoft Exchange clients.</td>
</tr>
<tr>
<td>Microsoft Exchange Mailbox</td>
<td>Exchange Client Access Server (cmdb_ci_exchange_cas)</td>
<td>• (For Exchange CAS 2010 and 2013, 2016) On the Now Platform, configure applicative credentials, setting the CI type to Microsoft Exchange Mailbox.</td>
</tr>
<tr>
<td>Microsoft Exchange CAS</td>
<td></td>
<td>Warning: Do not confuse the Exchange Mailbox CI type with the Microsoft Exchange Mailbox server CI type. Service Mapping uses these applicative credentials to access the of/Servers/Servers.slab folder on the Microsoft Exchange Admin Center (EAC). Service Mapping also uses the applicative credentials to run the following PowerShell commands against the Microsoft Exchange servers:</td>
</tr>
</tbody>
</table>
<pre><code>                                                                                 |                                                                                     | • Get-ExchangeServer |
                                                                                 |                                                                                     | • Get-ExchangeServer |
                                                                                 |                                                                                     | • Get-MailboxServer |
                                                                                 |                                                                                     | • Get-ClusteredMailboxServerStatus |
                                                                                 |                                                                                     | • Get-ExchangeServer |
                                                                                 |                                                                                     | • Get-StorageGroup |
                                                                                 |                                                                                     | • Get-MailboxDatabase |
</code></pre>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Internet Information Services (IIS)</td>
<td>Microsoft iis Web Server (cmdb_ci_microsoft_iis_web_server)</td>
<td>On the Now Platform, configure Windows credentials. (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Microsoft Internet Information Services (IIS) Virtual Directory</td>
<td>IIS Virtual Directory (cmdb_ci_iisdirectory)</td>
<td>On the Now Platform, configure Windows credentials. (Optional) To find ADO.NET connections, set the Credential alias attribute to (cmdb_ci_iisdirectory). Configuring this attribute allows you later to use the credential for this CI in patterns. Configure applicative credentials. (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Microsoft Message Queueing (MSMQ)</td>
<td>MSMQ (cmdb_ci_appl_msmq)</td>
<td>On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>Microsoft .NET Framework</td>
<td>.NET Application (cmdb_ci_appl_dot_net)</td>
<td>On the Now Platform, configure Windows credentials. (Optional) To find ADO.NET connections, set the Credential alias attribute to (cmdb_ci_iisdirectory). Configuring this attribute allows you later to use the credential for this CI in patterns. Provide an IIS Virtual Directory OS user with the rights to run the IIS service on Windows. Configure applicative credentials. (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Microsoft SharePoint</td>
<td>SharePoint (cmdb_ci_appl_sharepoint)</td>
<td>On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Set the <strong>Credential alias</strong> attribute to (cmdb_ci_appl_sharepoint). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) To find Microsoft SharePoint connections, configure Windows credentials for a user with permissions to run http get request to the SharePoint Central Administration URL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide a SharePoint OS user with permissions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To run SharePoint services on Windows</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To log into the SharePoint Admin page</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Configure applicative credentials.</strong></td>
</tr>
<tr>
<td>Microsoft SQL Database</td>
<td>MySQLClusterDataNode (cmdb_ci_db_mysql_clusternode)</td>
<td>Provide the following user or users:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For discovering Microsoft Biztalk using SQL queries, provide a user and credentials for Microsoft BizTalk.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For discovering licensing information on Microsoft SQL Server, provide a user and credentials for accessing Microsoft SQL Database.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Refer to the product documentation for operational information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Credentials for host discovery as described in Windows credentials.</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Configure applicative credentials.</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Unix server) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>----------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MySQL Cluster MGM Node</td>
<td>MySQLClusterMGMNode (cmdb_ci_db_mysql_clustermgmnode)</td>
<td>(If hosted on Unix only) Give user elevated rights to be able to run Service Mapping commands requiring a privileged user. On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>Nginx</td>
<td>Nginx Web Server (cmdb_ci_nginx_web_server)</td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user. configure applicative credentials; (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Oracle Clusterware</td>
<td>UNIX Cluster (cmdb_ci_unix_cluster), UNIX Cluster Node (cmdb_ci_unix_cluster_node), UNIX Cluster Resource (cmdb_ci_unix_cluster_resource), Cluster Virtual IPs (cmdb_ci_cluster_vip)</td>
<td>See <a href="#">Oracle database discovery</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run <a href="#">Service Mapping commands requiring a privileged user</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td>Oracle Concurrent Server</td>
<td>Oracle Concurrent Server (cmdb_ci_appl_ora_conc)</td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run <a href="#">Service Mapping commands requiring a privileged user</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>Oracle Instance (cmdb_ci_db_ora_instance)</td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run <a href="#">Service Mapping commands requiring a privileged user</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix) Configure <a href="#">applicative credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td>Oracle Database Advanced Queuing</td>
<td>Advanced Queue Queue (cmdb_ci_appl_ora_queue)</td>
<td>(If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Unix server) On the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Oracle Discoverer Engine</td>
<td>Oracle Discoverer Engine (cmdb_ci_appl_ora_disc)</td>
<td>Provide an Oracle instance user with permission to run the sqlplus command as described in Service Mapping commands not requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Oracle Discoverer UI</td>
<td>Oracle Discoverer UI (cmdb_ci_appl_ora_disc_ui)</td>
<td>(If hosted on Unix) Configure applicative credentials for Service Mapping.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix only) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Oracle E-Business Suite</td>
<td>Oracle ESB (cmdb_ci_appl_ora_ebs)</td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, configure a domain user for accessing the Windows OS as described in Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Oracle Form UI</td>
<td>Oracle Forms UI (cmdb_ci_appl_ora_forms_ui)</td>
<td>(Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure a domain user for accessing the Windows OS as described in <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oracle Fulfillment Server</td>
<td>Oracle Fulfillment Server (cmdb_ci_appl_ora_fs)</td>
<td>(Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Unix) Configure <a href="#">applicative credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oracle GlassFish Server</td>
<td>GlassFish (cmdb_ci_appl_glassfish) GlassFish WAR (cmdb_ci_appl_glassfish_war)</td>
<td>• (If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (If hosted on a Linux or Unix server) (Optional) on the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• (Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oracle HTTP Server</td>
<td>Oracle HTTP Server (cmdb_ci_appl_ora_http)</td>
<td>(If hosted on Unix) Configure <a href="#">applicative credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <a href="#">SSH credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see <a href="#">Configure WinRM trusted hosts on MID Server</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oracle IAS Web Module</td>
<td>Oracle IAS Web module (cmdb_ci_app_server_ora_ias_m)</td>
<td>(If hosted on Unix) Configure <a href="#">applicative credentials</a>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <a href="#">Windows credentials</a>.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Metric Client</td>
<td>Oracle Metric Client</td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_appl ora_metric_client)</td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>Oracle Metric Server</td>
<td>Oracle Metric Server</td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_appl ora_metric_svr)</td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Oracle MySQL Server</td>
<td>MySQL Instance</td>
<td>Provide the following user or users:</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_db mysql_instance)</td>
<td>• For discovering Microsoft Biztalk using SQL queries, provide a user and credentials for Microsoft BizTalk.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For discovering licensing information on Microsoft SQL Server, provide a user and credentials for accessing Microsoft SQL Database.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Configure applicative credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) Give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Linux server) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Oracle Net Listener</td>
<td>Oracle Database Listener</td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Oracle Notification Server</td>
<td>Oracle Notification Server</td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td>Oracle OACORE Server</td>
<td>Oracle OACORE Server</td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td>Oracle OAFM Server</td>
<td>Oracle OAFM Server</td>
<td>(If hosted on Unix) Configure applicative credentials.</td>
</tr>
<tr>
<td>Oracle PeopleSoft</td>
<td>Peoplesoft Application Server</td>
<td>(If hosted on Unix) On the Now Platform, configure SSH credentials.</td>
</tr>
<tr>
<td>Oracle Process Manager</td>
<td>Oracle Process Manager</td>
<td>Provide a user with permissions to run commands described in Service Mapping commands not requiring a privileged user.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>---------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see <strong>Configure WinRM trusted hosts on MID Server</strong>.</td>
</tr>
<tr>
<td>Oracle Tnslnr Engine</td>
<td>Oracle TNS Listener Engine (cmdb_ci_appl_ora_tnslsnr)</td>
<td>(If hosted on Unix) Configure <strong>applicative credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on Linux) On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(If hosted on a Windows server) On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
</tbody>
</table>
| Oracle Tuxedo and Oracle Tuxedo Portal        | Tuxedo (cmdb_ci_appl_tuxedo) and Tuxedo Portal (cmdb_ci_appl_tuxedo_portal) | • On the Now Platform, configure **SSH credentials**.  
|                                               |                                                   | • (Optional) For discovering using the WinRM protocol, see **Configure WinRM trusted hosts on MID Server**.  
<p>|                                               |                                                   | • (If hosted on Unix) Provide a user with the elevated rights for running commands described in <strong>Service Mapping commands requiring a privileged user</strong>. |
| Oracle WebLogic Module                        | Oracle iAS Web module (cmdb_ci_app_server_ora_ias_m) | (If hosted on Unix) Configure <strong>applicative credentials</strong>.                     |
| Oracle WebLogic Server (version 10.3)         | Weblogic (cmdb_ci_app_server_weblogic)            | (If hosted on Unix) Configure <strong>applicative credentials</strong>.                     |
|                                               |                                                   | (If hosted on a Windows server) On the Now Platform, configure <strong>Windows credentials</strong>. |
|                                               |                                                   | (If hosted on a Unix server) On the Now Platform, configure <strong>SSH credentials</strong>. |
|                                               |                                                   | (Optional) For discovering using the WinRM protocol, see <strong>Configure WinRM trusted hosts on MID Server</strong>. |
| Oracle WebLogic On-demand Router Load Balancer | Websphere ODR LB (cmdb_ci_app_server_ws_odr)      | (If hosted on Unix) Configure <strong>applicative credentials for Service Mapping</strong>. |
|                                               |                                                   | (If hosted on a Windows server) On the Now Platform, configure <strong>Windows credentials</strong>. |</p>
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radware Alteon NG</td>
<td>Alteon Load Balancer (cmdb_ci_lib_alteon)</td>
<td><em>(If hosted on a Unix server)</em> On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td>Radware AppDirector</td>
<td>Radware Load Balancer (cmdb_ci_lib_radware)</td>
<td>On the Now Platform, configure <strong>SNMP community credentials</strong>.</td>
</tr>
<tr>
<td>Red Hat Cluster Suite</td>
<td>Cluster Virtual IPs (cmdb_ci_cluster_vip)</td>
<td>On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td>UNIX Cluster (cmdb_ci_unix_cluster)</td>
<td>Provide a user with permissions to read these files:</td>
</tr>
<tr>
<td></td>
<td>UNIX Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>• /etc/cluster/cluster.conf</td>
</tr>
<tr>
<td></td>
<td>UNIX Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
<td>• /etc/hosts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide a user with elevated rights for running this command: clustat -x.</td>
</tr>
<tr>
<td>PostgreSQL Database</td>
<td>PostgreSQL Instance (cmdb_ci_db_postgresql_instance)</td>
<td>*(If hosted on Unix) Configure <strong>applicative credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Unix server)</em> On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(For hosted on Unix only)</em> Give this user elevated rights to be able to run <strong>Service Mapping commands requiring a privileged user</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>*(Optional) For discovering using the WinRM protocol, see <strong>Configure WinRM trusted hosts on MID Server</strong>.</td>
</tr>
<tr>
<td>Pivotal RabbitMQ</td>
<td>RabbitMQ Cluster (cmdb_ci_appl_rabbitmq_cluster)</td>
<td>*(If hosted on Unix) Configure <strong>applicative credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Windows server)</em> On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>(If hosted on a Unix server)</em> On the Now Platform, configure <strong>SSH credentials</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>*(Optional) For discovering using the WinRM protocol, see <strong>Configure WinRM trusted hosts on MID Server</strong>.</td>
</tr>
<tr>
<td>SAP Business Object XI Schedule</td>
<td>SAP BO BOXIScheduleRouter (cmdb_ci_appl_sap_bo_schedule)</td>
<td>On the Now Platform, configure <strong>Windows credentials</strong>.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>----------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SAP Crystal Management Server</td>
<td>SAP Business Objects CMS server (cmdb_ci_appl_sap_bo)</td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>----------------</td>
</tr>
</tbody>
</table>
| SAP Central Instance/ SAP Central Services (SCS)/ SAP Development Infrastructure/ SAP Evaluated Receipt Settlement (ERS)/ SAP Java Connector/ SAP NetWeaver Dialog Instance/ ABAP SAP Central Services (ASCS) | SAP CI Application (cmdb_ci_appl_sap_ci)/ SAP SCS Application (cmdb_ci_appl_sap_scs)/ SAP DI Application (cmdb_ci_appl_sap_di)/ SAP ERS Application (cmdb_ci_appl_sap_ers)/ SAP JC Application (cmdb_ci_appl_sap_jc)/ SAP DI Application (cmdb_ci_appl_sap_di)/ SAP ASCS Application (cmdb_ci_appl_sap_asc) | • Provide a user with permissions to run SOAP on RFC read table function.  
• (If hosted on a Unix server) On the Now Platform, configure [applicative credentials](https://www.sap.com/) and [SSH credentials](https://www.sap.com/). The SSH credentials are used to create connections from SAP Central Instance to other applications and devices. In addition, provide the user elevated rights to be able to run Service Mapping commands requiring a privileged user.  
• (If hosted on a Windows server) On the Now Platform, configure [Windows credentials](https://www.sap.com/) to create connections from SAP Central Instance to other applications and devices.  
• (Optional) For discovering using the WinRM protocol, see [Configure WinRM trusted hosts on MID Server](https://www.sap.com/).  
• (Optional) If your organization deploys SAP digital business services using web services and you want to map all connected business services, perform the following steps:  
  • Configure applicative credentials for a user with permissions to run the following commands: CVERS, DBCONS, and RFCONS.  
  • Ensure that you have the web service URL. For information on how to create web services, see [https://blogs.sap.com/2012/10/24/how-to-create-web-services-abap-based/](https://blogs.sap.com/2012/10/24/how-to-create-web-services-abap-based/).  
• (If hosted on a Windows server) On the Now Platform, configure [Windows credentials](https://www.sap.com/).  
• (If hosted on a Unix server) On the Now Platform, configure [SSH credentials](https://www.sap.com/). |
| SAP HANA Database | SAP Hana Db (cmdb_ci_appl_sap_hana_db) | |

© 2020 ServiceNow, Inc. All rights reserved.  
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Application</th>
<th>CI type</th>
<th>Configure this</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP Sybase DB</td>
<td>Sybase Instance (cmdb_ci_db_syb_instance)</td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. (If hosted on Unix) Configure applicative credentials. (If hosted on a Windows server) On the Now Platform, configure Windows credentials. (If hosted on a Unix server) On the Now Platform, configure SSH credentials. (For hosted on Unix only) Give this user elevated rights to be able to run Service Mapping commands requiring a privileged user. (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. Provide an SSRS OS user with permissions to run the SSRS Service. Configure credentials for a user with permissions to run http get request to the entry point URL. (If hosted on a Unix server) On the Now Platform, configure SSH credentials. Set the Credential alias attribute to (cmdb_ci_db_mssql_reporting). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td>SQL Server Integration Services (SSIS)</td>
<td>SQL Server Integration Services (cmdb_ci_db_mssql_integration)</td>
<td>On the Now Platform, configure Windows credentials.</td>
</tr>
<tr>
<td>SQL Server Reporting Services (SSRS)</td>
<td>SQL Server Reporting Services (cmdb_ci_db_mssql_reporting)</td>
<td>Provide an SSRS OS user with permissions to run the SSRS Service. Configure credentials for a user with permissions to run http get request to the entry point URL. (If hosted on a Unix server) On the Now Platform, configure SSH credentials. Set the Credential alias attribute to (cmdb_ci_db_mssql_reporting). Configuring this attribute allows you later to use the credential for this CI in patterns.</td>
</tr>
<tr>
<td>Application</td>
<td>CI type</td>
<td>Configure this</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Sun iPlanet Web Server</td>
<td>Iplanet Web Server (cmdb_ci_iplanet_web_server)</td>
<td>(Optional) To discover connections to report datasources, configure credentials for a user with the permission to run http post request to the entry point URL. Configure applicative credentials.</td>
</tr>
<tr>
<td>Sun Directory</td>
<td>Sun LDAP Server (cmdb_ci_sun_ldap_dir_server)</td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. Configure applicative credentials.</td>
</tr>
<tr>
<td>Sun JES</td>
<td>JES (cmdb_ci_email_server_jes)</td>
<td>(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server. Configure applicative credentials.</td>
</tr>
<tr>
<td>TIBCO EMS Queue/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIBCO Enterprise Message Service (EMS)/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIBCO ActiveMatrix Adapter</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Credentials required for host discovery

There are credentials and permissions that Service Mapping requires for discovering hosts.

Prior to discovering applications, Service Mapping accesses hosts that applications run on and discovers them. Make sure that you provide all necessary credentials to allow host discovery.

The Now Platform provides a centralized way of configuring credentials for Discovery, Service Mapping, and other applications. It allows you to configure credentials for hosts and applications only once. If you have already configured host-related credentials for another application, you do not need to do it again for Service Mapping.

Credentials required for discovery of hosts running on UNIX/Linux:

Discovery, Orchestration, and IntegrationHub explore UNIX and Linux devices by using SSH credentials to execute commands over Secure Shell (SSH).

To provide sufficient permissions, configure one of the following Unix and Linux credentials:

- Non-root user and password and using the `sudo` utility to run selected commands as root
- Root user and password

For information on commands requiring sudo-level rights, see Service Mapping commands requiring a privileged user and Commands that require root privileges for Discovery and Orchestration.

To access Unix-based hosts with non-root credentials, provide the read access to the following files and directories:

- /etc/*release
- /etc/bashrc
- /etc/profile
- /proc/cpuinfo
- /proc/vmware/sched/ncpus
- /var/log/dmesg
- APD directory

Credentials required for discovery of hosts running on Windows Servers:

To provide sufficient permissions, configure one of the following Windows credentials:

- A domain user with local administrator access on the target Windows hosts.
- A domain administrator.

Note: You may need domain administrator credentials only in some cases. For example, when discovering domain controllers.

Configure MID Server to use Windows credentials.

(Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.

Service Mapping commands requiring a privileged user

Service Mapping uses commands requiring elevated rights to discover and map Unix-based hosts in your organization. In addition to configuring necessary credentials, configure servers in your organization to allow Service Mapping to run these commands with elevated rights.
You do not run these commands directly. Service Mapping uses commands requiring elevated rights as part of the following processes:

- host detection
- process identification on port
- discovering CIs using patterns

Some commands are mandatory. The discovery process fails if Service Mapping cannot run these commands with elevated rights.

**Warning:** If you do not configure credentials for a mandatory command, a process or a pattern using it fails leading to discovery failure or error.

Some of these commands do not require elevated rights, unless directories that Service Mapping must access are protected.

### Operating system commands requiring elevated rights

#### AIX commands requiring elevated rights

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>file-name</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>ls</td>
<td>-F -1, -1HF, -w 1, -l</td>
<td>Mandatory</td>
<td>Lists the directory content.</td>
</tr>
<tr>
<td>procwdx</td>
<td>Process_id</td>
<td>Mandatory</td>
<td>Gets working directory of a process.</td>
</tr>
<tr>
<td>rmsock</td>
<td>Socketname tcpcp</td>
<td>It is mandatory to use either rmsock or lsof.</td>
<td>Finds process listening on a specific port.</td>
</tr>
<tr>
<td>lsof</td>
<td>-Pnl +M -l</td>
<td>It is mandatory to use either rmsock or lsof.</td>
<td>Shows files or connections associated with the process.</td>
</tr>
<tr>
<td>lstat</td>
<td>Various options</td>
<td>Optional</td>
<td>Fetches information about a link.</td>
</tr>
<tr>
<td>ps</td>
<td>eww</td>
<td>Mandatory</td>
<td>Fetches environment variables for the process on AIX.</td>
</tr>
</tbody>
</table>

#### HP-UX commands requiring elevated rights

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>file-name</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>ls</td>
<td>-F-1</td>
<td>Mandatory</td>
<td>Lists the directory content.</td>
</tr>
<tr>
<td></td>
<td>-1HF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-w 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-1HF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-w 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pfiles</td>
<td>Process_id</td>
<td>It is mandatory to use either pfiles or lsol.</td>
<td>Shows files or connections associated with the process.</td>
</tr>
<tr>
<td>lsol</td>
<td>-Pnl +M -i</td>
<td>It is mandatory to use either pfiles or lsol.</td>
<td>Shows files or connections associated with the process.</td>
</tr>
</tbody>
</table>

### Solaris commands requiring elevated rights

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>File_name</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>ls</td>
<td>ls -F-1</td>
<td>Mandatory</td>
<td>Lists the directory content.</td>
</tr>
<tr>
<td></td>
<td>ls -1HF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ls -w 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ls -1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>e-l</td>
<td>Optional</td>
<td>Fetches file attributes used for file caching decision.</td>
</tr>
<tr>
<td>dtrace</td>
<td>-C -s</td>
<td></td>
<td>Shows process listening on the given port. Required for Solaris 11.1 or earlier.</td>
</tr>
<tr>
<td></td>
<td>script_name</td>
<td>For Solaris 11.1 and earlier, it is mandatory to use either the dtrace or lsol command.</td>
<td></td>
</tr>
<tr>
<td>lsol</td>
<td>-Pnl +M -i</td>
<td>For Solaris 11.1 and earlier, it is mandatory to use either the dtrace or lsol command.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>For Solaris 11.2 and later, it is mandatory to use either the lsol or netstat command.</td>
<td></td>
</tr>
<tr>
<td>netstat</td>
<td>-anu</td>
<td>For Solaris 11.2 and later, it is mandatory to use either the lsol or netstat command.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lists the open ports. Required for Solaris version 11.2 or later.</td>
<td></td>
</tr>
<tr>
<td>ifconfig</td>
<td>ifconfig -a</td>
<td>Mandatory</td>
<td>Shows interface information (need sudo to get the MAC addresses).</td>
</tr>
<tr>
<td>pwdx</td>
<td>Process_id</td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
<tr>
<td>pargs</td>
<td>-e process_id</td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
</tbody>
</table>
### Command Parameter Mandatory/optional Description

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ps</td>
<td>-eo user, pid, ppid, comm, args</td>
<td>Mandatory</td>
<td>Gets the process list.</td>
</tr>
<tr>
<td>inetadm</td>
<td>-l or without params</td>
<td>Optional</td>
<td>Handles the case of application using the inet daemon.</td>
</tr>
</tbody>
</table>

### Linux commands requiring elevated rights

<table>
<thead>
<tr>
<th>Commands</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>File_name</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>ls</td>
<td>ls -F-1</td>
<td>Mandatory</td>
<td>Lists directory content.</td>
</tr>
<tr>
<td></td>
<td>ls -1HF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ls -w 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ls -1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>netstat</td>
<td>-lnup</td>
<td>Mandatory</td>
<td>Shows the open network connections.</td>
</tr>
<tr>
<td></td>
<td>-lnp</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-ntup</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-an</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stat</td>
<td>--format=&quot;%Y&quot;</td>
<td>Optional</td>
<td>Shows the modification time of the file.</td>
</tr>
<tr>
<td></td>
<td>-L file_name</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Application commands requiring elevated rights

Service Mapping uses some of these commands in patterns.

**Apache Web Server (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory only for the OC4J Connectivity - netstat - linux connection section of the pattern. If elevated rights are not configured for this command, Service Mapping does not discover Apache Web Server connections to Oracle Application Server Containers for J2EE.</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Is</td>
<td>Various options</td>
<td>Mandatory only for the Create Weblogic Connections connection section of the pattern. If elevated rights are not configured for this command, Service Mapping does not discover Apache Web Server connections to Oracle WebLogic Module.</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

**Apache Tomcat (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only for creating the web services connections. Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**Apache Tomcat WAR (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>version.sh/version.bat</td>
<td>-</td>
<td>Optional</td>
<td>Gets the Tomcat version.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>find</strong></td>
<td>-name</td>
<td>Optional</td>
<td>This command is used in connection sections of the pattern for discovering Apache Tomcat WAR connections. Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**App TNS Service (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

**BIG-IP Global Traffic Manager (GTM) F5 (on F5 BIG-IP)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ping</td>
<td>-a -c 1 (url1).host</td>
<td>Mandatory</td>
<td>Gets the host IP.</td>
</tr>
</tbody>
</table>

**BMC CTRL-M Enterprise Manager (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**BMC CTRL-M Gateway (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only to create connections to Control M Server. Finds file and folder paths.</td>
</tr>
</tbody>
</table>
### Citrix Netscaler (on Citrix Netscaler)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>show</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Retrieves Netscaler IP address.</td>
</tr>
</tbody>
</table>

### EMC Documentum Broker (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>Finds connections to Docbase.</td>
</tr>
</tbody>
</table>

### FormEngine (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>(On Unix only) Shows the file content.</td>
</tr>
</tbody>
</table>

### IBM Customer Information Control System (CICS) (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

### IBM CTRL-M Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Finds file and folder paths.</td>
</tr>
</tbody>
</table>
### IBM CICS Transaction Gateway CTG (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Optional</td>
<td>(For Unix only) This command is used only to create CICS connections. Extracts strings from the output.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Optional</td>
<td>(For Unix only) This command is used only to create CICS connections. Displays the file content in the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>(For Unix only) This command is used only to create CICS connections. Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### IBM DB2 (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the file content in the output.</td>
</tr>
</tbody>
</table>

### IBM J2EE EAR (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Mandatory</td>
<td>Finds files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

### IBM Websphere Application Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>find</code></td>
<td><code>-name</code></td>
<td>Optional</td>
<td>This command is used only for creating the Web Services connections. Finds files and folders for the specific name.</td>
</tr>
<tr>
<td></td>
<td><code>-type</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### IBM Websphere Portal (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ls</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td><code>cat</code></td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td><code>netstat</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

### IBM WebSphere Message Broker Flow (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>grep</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td><code>ls</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td><code>cat</code></td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td><code>ps</code></td>
<td><code>-ef</code></td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
<tr>
<td><code>netstat</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

### IBM WebSphere Message Broker (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ps</code></td>
<td><code>-ef</code></td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
<tr>
<td><code>netstat</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
<tr>
<td><code>grep</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td><code>ls</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td><code>cat</code></td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### IBM WebSphere MQ (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>grep</code></td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>
### IBM WebSphere MQ Queue (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

### Jboss Module (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### Jboss (on Unix and Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### JRun (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

### JRun WAR Inc (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>
### Microsoft Exchange Client Access Server (CAS), Microsoft Exchange Hub Transport Server, Microsoft Exchange Mailbox (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get-ExchangeServer</td>
<td>-status -Identity &quot;+ $hostname+&quot;</td>
<td>Mandatory for using PowerShell commands.</td>
<td>Extracts the list of Microsoft Exchange hosts and role</td>
</tr>
<tr>
<td>Get-ExchangeServer</td>
<td>format-table -autosize -HideTableHeaders Name,Fqdn,IsMailboxServer</td>
<td>Mandatory for using PowerShell commands.</td>
<td>Extracts the list of Microsoft Exchange hosts and role in table format</td>
</tr>
<tr>
<td>Get-MailboxServer</td>
<td>format-list</td>
<td>Mandatory for using PowerShell commands.</td>
<td>Extracts the list of Microsoft Exchange Mailbox servers.</td>
</tr>
<tr>
<td>Get-ClusteredMailBoxServerStatus</td>
<td>format-table -Property OperationalMachines</td>
<td>Mandatory for using PowerShell commands.</td>
<td>Extracts information about Microsoft Exchange Mailbox clusters.</td>
</tr>
<tr>
<td>Get-StorageGroup</td>
<td>-server &quot;+Shname+&quot;</td>
<td>Mandatory for discovering storage.</td>
<td>Extracts information about storage and exports it into a file.</td>
</tr>
</tbody>
</table>

- ** grep **
  * Various options
  * Mandatory
  * (On Unix only) Extracts strings from the output.

- ** netstat **
  * Various options
  * Mandatory
  * Lists open ports.

- ** ps **
  * --pid=(process.pid) --no-headers -o %U : %p : %a
  * Optional
  * Gets the userid parameter value.
### MySQL Cluster MGM Node (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>ps</td>
<td>--pid=(process,pid) --no-headers -o &quot;%U : %p : %a&quot;</td>
<td>Optional</td>
<td>(On Unix only) Gets the userid parameter value.</td>
</tr>
</tbody>
</table>

### Nginx (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>ps</td>
<td>--pid=(process,pid) --no-headers -o &quot;%U : %p : %a&quot;</td>
<td>Optional</td>
<td>Gets the userid parameter value.</td>
</tr>
<tr>
<td>ls</td>
<td>(IncludeTabletmp().files)</td>
<td>Optional</td>
<td>This command is used if necessary to create HTTP connections. Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>(IncludeTabletmp().files)</td>
<td>Optional</td>
<td>This command is used if necessary to create HTTP connections. Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle E-Business Suite (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>export</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets variables.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>sqlplus</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Creates the connection to the Oracle instance.</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Manipulates the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Gets open ports.</td>
</tr>
</tbody>
</table>

### Oracle Advanced Queue Queue (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>
### Oracle Concurrent Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Manipulates the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>(For Unix only) Displays the file content in the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Oracle Discoverer Engine (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle Discoverer UI (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle Form UI (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>
### Oracle Fulfillment Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle HTTP Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle Metric Client (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle OACORE Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

### Oracle OAFM Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>
Oracle Database (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>ps</td>
<td>-ef</td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

Oracle iAS Web Module (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>(For Unix only) Shows the file content.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

Oracle Process Manager (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

Oracle Report Server (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>
## Oracle Tnslsnr Engine (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

## Oracle WebLogic Module (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only for creating the Web Services connections. Finds files and folders for the specific name.</td>
</tr>
<tr>
<td></td>
<td>-type</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Oracle WebLogic Server (version 10.3) (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

## Oracle WebLogic On-demand Router Load Balancer (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
</tbody>
</table>

## Oracle Tuxedo (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ls</td>
<td>Various</td>
<td>Mandatory</td>
<td>Lists files and folders on the given folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the file content in the output.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only to create web service connections. Finds files and folders for specific name.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various</td>
<td>Mandatory</td>
<td>Gets open ports.</td>
</tr>
<tr>
<td>ps</td>
<td>-ef</td>
<td></td>
<td>(For Unix only) Gets process attributes.</td>
</tr>
</tbody>
</table>

**Oracle Tuxedo Portal (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>ls</td>
<td>Various</td>
<td>Mandatory</td>
<td>Lists files and folders on the given folder.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the file content in the output.</td>
</tr>
</tbody>
</table>

**PostgreSQL Database (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>(IncludeTabletmp().files)</td>
<td>Optional</td>
<td>This command is used only to create the HTTP connections. Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

**Rabbit MQ (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ps - ef</td>
<td>ps -ef</td>
<td>Mandatory</td>
<td>Gets the parent process.</td>
</tr>
</tbody>
</table>

**Red Hat Cluster Suite (on Linux)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>clustat</td>
<td>-x</td>
<td>Mandatory</td>
<td>Displays the cluster configuration and the status in XML format.</td>
</tr>
</tbody>
</table>
### SAP components including ABAP SAP Central Services (ASCS), SAP Control Instance, SAP Netweaver Dialog Instance, SAP Evaluated Receipt Settlement (ERS), SAP Java Cluster (SJC), SAP Central Services (SCS) (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>ping</td>
<td>-c 1</td>
<td>Mandatory</td>
<td>Verifies that the host is answering.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>CVERS</td>
<td>-</td>
<td>Optional</td>
<td>Retrieves the version of installed SAP modules.</td>
</tr>
<tr>
<td>DBCONS</td>
<td>-</td>
<td>Optional</td>
<td>Retrieves the connection strings to SAP DB.</td>
</tr>
<tr>
<td>RFCDES</td>
<td>-</td>
<td>Optional</td>
<td>Retrieves the connection string to systems integrated with SAP.</td>
</tr>
</tbody>
</table>

#### Sun Directory (on Solaris)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

#### Sun iPlanet Web Server (on Solaris)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

#### Sun JES (on Solaris)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

#### Sybase (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>
### TIBCO ActiveMatrix Adapter (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### TIBCO ActiveMatrix BusinessWorks (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### TIBCO ActiveMatrix BusinessWorks Process (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>cut</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Splits the output line.</td>
</tr>
</tbody>
</table>

### TIBCO EMS Queue (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

### Tibco Enterprise Message Service (EMS) (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>
Service Mapping commands not requiring a privileged user

Most of the commands utilized by Service Mapping for discovery and mapping do not require elevated rights.

Review this list of commands to understand how Service Mapping uses them and to make sure that the virtual security of your company is not compromised.

You do not run these commands directly. Service Mapping uses commands requiring elevated rights as part of the following processes:

- host detection
- process identification on port
- discovering CIs using patterns

Some commands are mandatory. The discovery process fails if Service Mapping cannot run these commands.

**Operating systems**

**AIX commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>awk</td>
<td>various options</td>
<td>Mandatory</td>
<td>Parses the string.</td>
</tr>
<tr>
<td>istat</td>
<td>file_name</td>
<td>Optional</td>
<td>Gets last modification time of file.</td>
</tr>
<tr>
<td>nslookup</td>
<td>hostname</td>
<td>Mandatory</td>
<td>Resolve DNS host name.</td>
</tr>
<tr>
<td>ping</td>
<td>-c count ip/host</td>
<td>Mandatory</td>
<td>Pings the host.</td>
</tr>
<tr>
<td>arp -an</td>
<td>various options</td>
<td>Mandatory</td>
<td>Shows ARP table.</td>
</tr>
<tr>
<td>grep</td>
<td>String to search</td>
<td>Mandatory</td>
<td>Finds string in previous command output.</td>
</tr>
<tr>
<td>traceroute</td>
<td>-n ip_address</td>
<td>Mandatory</td>
<td>Shows layer 3 network route to target host.</td>
</tr>
</tbody>
</table>
| netstat | -an  
         | -Aan | Mandatory | Shows open network connections. |
| ps      | -eo user, pid, ppid, comm, args | Mandatory | Gets the process list. |
| grep    | various options | Mandatory | Searches text in previous command output. |
| find    | Directory and parameters to search | Mandatory | Searches for files by name or type. |

**HP-UX commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ping</td>
<td>-c count ip/host</td>
<td>Mandatory</td>
<td>Pings the host.</td>
</tr>
<tr>
<td>arp -an</td>
<td></td>
<td>Optional (required by network discovery)</td>
<td>Shows the ARP table.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------------------------</td>
<td>--------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>grep</td>
<td>String to search</td>
<td>Mandatory</td>
<td>Finds string in previous command output.</td>
</tr>
<tr>
<td>awk</td>
<td>various options</td>
<td>Mandatory</td>
<td>Parses the string.</td>
</tr>
<tr>
<td>traceroute</td>
<td>-n ip_address</td>
<td>Mandatory</td>
<td>Shows layer 3 network route to target host.</td>
</tr>
<tr>
<td>netstat</td>
<td>-an</td>
<td>Mandatory</td>
<td>Shows open network connections.</td>
</tr>
<tr>
<td>ps</td>
<td>-eo user, pid, ppid, comm, args</td>
<td>Mandatory</td>
<td>Gets the process list.</td>
</tr>
<tr>
<td>grep</td>
<td>various options</td>
<td>Mandatory</td>
<td>Searches text in previous command output.</td>
</tr>
<tr>
<td>ps</td>
<td>-eo user, pid, ppid, comm, args</td>
<td>Mandatory</td>
<td>Gets the process list.</td>
</tr>
<tr>
<td>find</td>
<td>Directory and parameters for search</td>
<td>Mandatory</td>
<td>Searches for files by name or type.</td>
</tr>
</tbody>
</table>

**IBM z/OS**

<table>
<thead>
<tr>
<th>Commands</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sysvar</td>
<td>SYSNAME</td>
<td>Mandatory</td>
<td>Gets the system name.</td>
</tr>
<tr>
<td>hostname</td>
<td></td>
<td>Mandatory</td>
<td>Retrieves the device name.</td>
</tr>
<tr>
<td>grep</td>
<td>String to search</td>
<td>Mandatory</td>
<td>Finds the string in command output.</td>
</tr>
<tr>
<td>hostname</td>
<td></td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>netstat</td>
<td>-g</td>
<td>awk '{if (NR&gt;4) print $1, $3}'</td>
<td>Mandatory</td>
</tr>
<tr>
<td>netstat</td>
<td>-r</td>
<td>awk '{if (NR&gt;3) print $1, $2, $5}'</td>
<td>Mandatory</td>
</tr>
<tr>
<td>netstat</td>
<td>-h</td>
<td>awk '{if (NR&gt;4) print $1}</td>
<td>grep -v '127.0.0.1'</td>
</tr>
<tr>
<td>netstat</td>
<td>-R</td>
<td>ALL</td>
<td>awk '{if (NR&gt;1) print $0}'</td>
</tr>
<tr>
<td>netstat</td>
<td>df -k</td>
<td>awk '{print $1, $2}</td>
<td>sed -e s/0/</td>
</tr>
<tr>
<td>ps</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the process list.</td>
</tr>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>cat</td>
<td>file-name</td>
<td>Mandatory</td>
<td>Shows the file content.</td>
</tr>
<tr>
<td>cut</td>
<td>Various options.</td>
<td>Mandatory</td>
<td>Splits the output according to entered parameters.</td>
</tr>
<tr>
<td>Commands</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>uname</td>
<td>l, v, rm, s</td>
<td>Mandatory</td>
<td>Gets the OS type.</td>
</tr>
</tbody>
</table>

**Linux**

<table>
<thead>
<tr>
<th>Commands</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nslookup</td>
<td>hostname</td>
<td>Mandatory</td>
<td>Resolves the DNS host name.</td>
</tr>
<tr>
<td>ping</td>
<td>-c count ip/host</td>
<td>Mandatory</td>
<td>Pings the host.</td>
</tr>
<tr>
<td>arp</td>
<td>-an</td>
<td>Optional (required for network path calculations)</td>
<td>Shows the ARP table.</td>
</tr>
<tr>
<td>grep</td>
<td>String to search</td>
<td>Mandatory</td>
<td>Finds the string in previous command output.</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Parses the string.</td>
</tr>
<tr>
<td>netstat</td>
<td>-an</td>
<td>Mandatory</td>
<td>Shows open network connections.</td>
</tr>
<tr>
<td>ps</td>
<td>-eo user, pid, ppid, comm, args</td>
<td>Mandatory</td>
<td>Gets the process list.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Searches text in previous command output.</td>
</tr>
<tr>
<td>egrep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Searches text in previous command output.</td>
</tr>
<tr>
<td>find</td>
<td>Directory and parameters for search</td>
<td>Mandatory</td>
<td>Searches for files by name or type.</td>
</tr>
<tr>
<td>traceroute</td>
<td>-n ip_address</td>
<td>Optional (required for network path calculation)</td>
<td>Shows the layer 3 network route to target host.</td>
</tr>
</tbody>
</table>

**Solaris**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nslookup</td>
<td>hostname</td>
<td>Mandatory</td>
<td>Resolves the DNS host name.</td>
</tr>
<tr>
<td>ping</td>
<td>-c count ip/host</td>
<td>Mandatory</td>
<td>Pings the host.</td>
</tr>
<tr>
<td>arp</td>
<td>-an</td>
<td>Optional (required for network path calculation)</td>
<td>Shows the ARP table.</td>
</tr>
<tr>
<td>grep</td>
<td>String to search</td>
<td>Mandatory</td>
<td>Finds the string in previous command output.</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Parses the string.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------</td>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>traceroute</td>
<td>-n ip_address</td>
<td>Optional (required for network path calculation)</td>
<td>Shows the layer 3 network route to target host.</td>
</tr>
<tr>
<td>netstat</td>
<td>-an</td>
<td>Mandatory</td>
<td>Shows the open network connections.</td>
</tr>
<tr>
<td>zoneadm</td>
<td>List</td>
<td>Mandatory</td>
<td>Shows the list of zones.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Searches text in previous command output.</td>
</tr>
<tr>
<td>egrep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Searches text in previous command output.</td>
</tr>
<tr>
<td>find for search</td>
<td>Directory and parameters</td>
<td>Mandatory</td>
<td>Searches for files by name or type.</td>
</tr>
</tbody>
</table>

### Windows WMI Queries

<table>
<thead>
<tr>
<th>Query</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select * from Win32_ComputerSystem</td>
<td>Gets the server basic information like serial number.</td>
</tr>
<tr>
<td>Select LastModified from CIM_LogicalFile Where Name=...</td>
<td>Gets the last modification time of a file.</td>
</tr>
<tr>
<td>Select AddressWidth from Win32_Processor</td>
<td>Gets the computer architecture (32 bit or 64 bit).</td>
</tr>
<tr>
<td>Select * From Win32_Process where (ProcessId = ?)</td>
<td>Gets the process list running on the computer.</td>
</tr>
<tr>
<td>Select Name from CIM_LogicalDisk</td>
<td>Gets the file systems on the computer, such as C: or D:.</td>
</tr>
<tr>
<td>Select FileName, Extension from CIM_Directory where Drive='drive' and Path='path'</td>
<td>Gets the list of files in a specific directory.</td>
</tr>
</tbody>
</table>

### Windows commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_process_info.exe</td>
<td>Process_id</td>
<td>Mandatory</td>
<td>Extracts information on the processes using executable created by ServiceNow.</td>
</tr>
<tr>
<td>type</td>
<td>File_name</td>
<td>Mandatory</td>
<td>Shows the text file content.</td>
</tr>
<tr>
<td>nslookup</td>
<td>Host_name</td>
<td>Mandatory</td>
<td>DNS lookup</td>
</tr>
<tr>
<td>arp</td>
<td>-a</td>
<td>Optional (required for network path calculation)</td>
<td>Shows the ARP table.</td>
</tr>
<tr>
<td>netstat</td>
<td>-ano</td>
<td>Mandatory</td>
<td>Shows the network connections.</td>
</tr>
</tbody>
</table>
### Command | Parameter | Mandatory/optional | Description
--- | --- | --- | ---
findstr | Various options | Mandatory | Finds the string in the previous command output.
dir | /q /s /b | Mandatory | Lists files in the directories.
paping | --nocolor -c 1 -p port_num ip_address | Mandatory | Utility to run TCP ping against given host.
ping | -n 1 ip_address/host | Mandatory | Pings the host.
tracert | ip_address | Optional (required for network path calculation) | Shows the layer 3 network route to the target host.

---

**Applications**

**ABAP SAP Central Services (ASCS) (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
</table>
dir | Various options | Mandatory | (For Windows only) Lists files and folders in the specified folder.
msg_server | -V | Optional | Gets the version.
ping | -n 1 | Mandatory | (For Windows only) Verifies that the host is answering.
netstat | Various options | Mandatory | Lists open ports.

**Active Directory Domain Controller (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
</table>
netstat | Various options | Mandatory | Gets open ports on
hostname | - | Mandatory | Gets the hostname.
dir | Various options | Mandatory | Lists all files in the specified folder.

**Apache Web Server (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/optional</th>
<th>Description</th>
</tr>
</thead>
</table>
httpd | -V | Optional | Gets the version.
cut | Various options | Mandatory | Splits the output line.
<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>opmnctl</td>
<td>status -fmt %cmp32%prt32%por40%pid and @farm status -noheaders -fsep &quot;</td>
<td>Optional</td>
<td>This command is used only for discovering the OC4J connectivity. Gets the status of the OPMN CTRL service.</td>
</tr>
<tr>
<td></td>
<td>&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-fmt %cmp%prt%clu%ins %por</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sort</td>
<td>-u</td>
<td>Optional</td>
<td>This command is used only for discovering the Weblogic connectivity. Sorts the output.</td>
</tr>
</tbody>
</table>

**Apache Web Server (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>httpd.exe</td>
<td>-V</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>opmnctl</td>
<td>status -fmt %cmp32%prt32%por40%pid and @farm status -noheaders -fsep &quot;</td>
<td>Optional</td>
<td>This command is used only for discovering the OC4J connectivity. Gets the status of the OPMN CTRL service.</td>
</tr>
<tr>
<td></td>
<td>&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-fmt %cmp%prt%clu%ins %por</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Apache Tomcat (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>version.sh/version.bat</td>
<td>-</td>
<td>Optional</td>
<td>Gets the Tomcat version.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only for creating the web services connections. Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Apache Tomcat WAR (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>version.sh/version.bat</td>
<td>-</td>
<td>Optional</td>
<td>Gets the Tomcat version.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(Windows only) Lists the open ports.</td>
</tr>
</tbody>
</table>

### App TNS Service (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists open ports.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
</tbody>
</table>

### BIG-IP Global Traffic Manager (GTM) F5 (on F5 BIG-IP)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>/gtm pool (wide_ip_pool_4_cmd[1]) members</td>
<td>Optional</td>
<td>This command is used only to create connections to pool members. Gets the pool members.</td>
</tr>
</tbody>
</table>
### BIG-IP Local Traffic Manager (LTM) F5 (on F5 BIG-IP)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>list</td>
<td>/gtm server (servers_from_cmd().server addresses)</td>
<td>Optional</td>
<td>This command is used only to create connections to pool members. Lists the alias server addresses.</td>
</tr>
</tbody>
</table>

**BIG-IP Device Service Clustering**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show cm traffic-group – get</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used to get traffic and clustering information.</td>
</tr>
</tbody>
</table>

### BMC CTRL-M Enterprise Manager (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists open ports.</td>
</tr>
</tbody>
</table>

### BMC CTRL-M Gateway (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------</td>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only to create connections to Control M Server. Finds file and folder paths.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xargs</td>
<td>grep &lt;string&gt;</td>
<td>Optional</td>
<td>This command is used only to create connections to IBM CTRL-M Server. Runs the command on all lists from the output.</td>
</tr>
</tbody>
</table>

**BMC IT Asset Management (ITAM) Software**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tnsping</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Retrieves information about Oracle connections.</td>
</tr>
</tbody>
</table>

**CA Identity Manager Provisioning Server (for Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

**Cisco ACE Command Line Interface (on Cisco CSM)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>context, running</td>
<td>Mandatory</td>
<td>Shows the requested information.</td>
</tr>
<tr>
<td>include</td>
<td>-</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>echo</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays strings to the output.</td>
</tr>
</tbody>
</table>

**Cisco Global Site Selector (on Cisco CSM)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>gslb-config domain-list</td>
<td>Mandatory</td>
<td>Shows the domain list.</td>
</tr>
<tr>
<td>show</td>
<td>gslb-config dns rule</td>
<td>Mandatory</td>
<td>Shows the DNS rule output.</td>
</tr>
<tr>
<td>show</td>
<td>gslb-config answer-group</td>
<td>Mandatory</td>
<td>Show the VIP answer.</td>
</tr>
</tbody>
</table>
### Citrix Delivery Controller (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>powershell</td>
<td>Add-PSSnapin Citrix.Broker.Admin.V2; Get-BrokerApplication -Name (entry_point.name) I select Name,CommandLineExecutable,CommandLineArguments,WorkingDirectory,ApplicationType,BrowserName</td>
<td>Mandatory</td>
<td>Gets applications managed by this Delivery Controller.</td>
</tr>
<tr>
<td></td>
<td>And Add-PSSnapin Citrix.Broker.Admin.V2; Get-BrokerApplication -Name (entry_point.name) I select Name,CommandLineExecutable,CommandLineArguments,WorkingDirectory,ApplicationType,BrowserName</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>And dd-PSSnapin Citrix.Broker.Admin.V2; Get-BrokerMachine -DesktopGroupName ` (delivery_groups(1).name)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Citrix Usage Collector (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qfarm</td>
<td></td>
<td>Mandatory</td>
<td>Discovers epic icons.</td>
</tr>
</tbody>
</table>

### Citrix XenApp (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_xenapp_apps.ps1</td>
<td>(entry_point.icon_path)</td>
<td>Mandatory</td>
<td>Runs powershell commands that retrieve the Citrix icon info from the Citrix repository.</td>
</tr>
<tr>
<td>powershell</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Runs powershell commands.</td>
</tr>
</tbody>
</table>

### Citrix Presentation Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cscript</td>
<td>//NoLogo</td>
<td>Mandatory</td>
<td>Runs VB scripts without a popup box.</td>
</tr>
<tr>
<td>GetAppsInFolder.wsf</td>
<td>(entry_point.icon_path)</td>
<td>Mandatory</td>
<td>Runs vbscript commands that retrieve the Citrix icon info from the Citrix repository.</td>
</tr>
</tbody>
</table>
### Citrix Netscaler (on Citrix Netscaler)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>cs policy, cs action, cs Policy, lb vserver</td>
<td>Mandatory</td>
<td>This command is used only for discovering outgoing connections. Shows the requested information.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

### Citrix Netscaler Global Server Load Balancer (on Citrix Netscaler)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>ns.conf</td>
<td>Mandatory</td>
<td>This command is used only for discovering outgoing connections if SNMP or SSH credentials are not provided. Shows the contents of the ns.conf file.</td>
</tr>
</tbody>
</table>

### Connect-IT Service (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

### EMC Documentum Docbase (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### FormEngine (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>
### Generic application (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>tasklist</td>
<td>Various options</td>
<td>Optional</td>
<td>Lists all running tasks.</td>
</tr>
<tr>
<td>netsh</td>
<td>http show servicestate</td>
<td>Optional</td>
<td>Shows a snapshot of the HTTP service.</td>
</tr>
</tbody>
</table>

### Generic application (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>systemctl list-units --type service</td>
<td>Various options</td>
<td>Optional</td>
<td>Show the service status.</td>
</tr>
<tr>
<td>/bin/systemctl status *+.service</td>
<td>Various options</td>
<td>Optional</td>
<td>Show the service status.</td>
</tr>
<tr>
<td>service --status-all</td>
<td>Various options</td>
<td>Optional</td>
<td>Show the service status.</td>
</tr>
<tr>
<td>read /etc/services</td>
<td>Various options</td>
<td>Optional</td>
<td>Allows access to the file.</td>
</tr>
</tbody>
</table>

### HAPrxy (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ls</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>tail</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Displays the end of the output.</td>
</tr>
</tbody>
</table>

### HP Service Manager Application Server

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tnsping</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Retrieves information about Oracle connections.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>
### HP Service Manager Index (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### HP Service Manager Knowledge Base (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### IBM WebSphere Message Broker (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dmqdocbroker</td>
<td>-s -c getdocbasemap and -t (computer_system.primaryHostname)</td>
<td>Mandatory</td>
<td>Gets the docbase information.</td>
</tr>
<tr>
<td></td>
<td>-s -c getdocbasemap and -t (computer_system.primaryHostname)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-s -c getservermap (docbas)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-t (computer_system.primaryHostname)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-s -c getdocbasemap (docbase)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>del</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Deletes the XML file containing</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the old binding information.</td>
</tr>
<tr>
<td>BTSTask</td>
<td>ExportBindings / Destination:%TEMP %\Mybindings.xml / Database: (MgmtDbName) / server:(serverName)</td>
<td>Optional</td>
<td>This command is used only if</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>there are not MSSQL credentials.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Extracts the binding info</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and places it into the XML file.</td>
</tr>
</tbody>
</table>

### IBM CTRL-M Server (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>
### IBM CICS Transaction Gateway CTG (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Optional</td>
<td>(For Windows only) This command is used only to create CICS connections. Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Optional</td>
<td>(For Windows only) This command is used only to create CICS connections. Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>type</td>
<td>-</td>
<td>Optional</td>
<td>(For Windows only) This command is used only to create CICS connections. Displays the file content in the output.</td>
</tr>
</tbody>
</table>

### IBM DB2 (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>(installed_dir)db2profile</td>
<td>Mandatory</td>
<td>Sets the DB2 variables.</td>
</tr>
<tr>
<td>db2</td>
<td>list database directory</td>
<td>Mandatory</td>
<td>Displays the DB2 installation parameters.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>head</td>
<td>-1</td>
<td>Mandatory</td>
<td>Displays only the first line from the output.</td>
</tr>
<tr>
<td>db2level</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the DB2 version.</td>
</tr>
<tr>
<td>cat</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the file content in the output.</td>
</tr>
<tr>
<td>xargs</td>
<td>-I {} echo list tablespace containers for {} ;©</td>
<td>Optional</td>
<td>This command is used only to create storage connections. Runs the command on all lists from the output.</td>
</tr>
</tbody>
</table>

### IBM DB2 (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>db2cmd</td>
<td>/c /w /i</td>
<td>Mandatory</td>
<td>Lists the DB2 directories.</td>
</tr>
<tr>
<td>db2</td>
<td>list database directory</td>
<td>Mandatory</td>
<td>Displays the DB2 installation parameters.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files, folders, or standard output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>db2level</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the DB2 version.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only for storage connectivity. Prints the strings in the output.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only for storage connectivity. Extracts strings from the output.</td>
</tr>
<tr>
<td>for</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only for storage connectivity. Runs loops.</td>
</tr>
</tbody>
</table>

**IBM DB2 (on IBM z/OS)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Gets the DB2 name.</td>
</tr>
</tbody>
</table>

**IBM J2EE EAR (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

**IBM Websphere Application Server (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>versionInfo.sh</td>
<td>-</td>
<td>Optional</td>
<td>Gets the Websphere Application Server version.</td>
</tr>
</tbody>
</table>
IBM Websphere Application Server (on IBM z/OS)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`oeconsol 'D OMVS,A=ALL</td>
<td>grep '+ $taskname`</td>
<td>-</td>
<td>Mandatory</td>
</tr>
<tr>
<td>netstat</td>
<td>-a</td>
<td>grep '+$port+'</td>
<td>Mandatory</td>
</tr>
<tr>
<td>ps -ef-o user,pid,ppid,comm</td>
<td>awk '{print $1}'</td>
<td>Mandatory</td>
<td>Gets attributes for processes.</td>
</tr>
<tr>
<td>`$install_dir+/bin/ versionInfo.sh</td>
<td>grep Version</td>
<td>tail -1`</td>
<td>Optional, if <code>$install_dir+/properties/version/ BASE.product</code> command brings output.</td>
</tr>
<tr>
<td><code>$install_dir+/properties/ version/BASE.product</code></td>
<td>Optional, if `$install_dir+/bin/ versionInfo.sh</td>
<td>grep Version</td>
<td>tail -1<code>.ophysical, if </code>$install_dir+/properties/version/ BASE.product` command brings output.</td>
</tr>
<tr>
<td>`ls &quot;$conf_dir&quot;<em>/applications/</em>/deployments/*/META-INF/application.xml</td>
<td>xargs grep &quot;&lt;context-root&gt;/&quot;+$url_search +&quot;&quot;</td>
<td>Optional, if `$install_dir+/bin/ versionInfo.sh</td>
<td>grep Version tail -1<code>.ophysical, if </code>$install_dir+/properties/version/ BASE.product` command brings output.</td>
</tr>
<tr>
<td><code>$ear_directory_name+/*/wsdl/*.wsdl</code></td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
<td></td>
</tr>
</tbody>
</table>

IBM Websphere Portal (on Linux)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>-name -type</td>
<td>Optional</td>
<td>This command is used only for creating the Web Services connections. Finds files and folders for the specific name.</td>
</tr>
</tbody>
</table>

IBM Websphere Portal (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**IBM WebSphere Message Broker Flow (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td>-name -type</td>
<td>Optional</td>
<td>This command is used only for creating the Web Services connections. Finds files and folders for the specific name.</td>
</tr>
<tr>
<td>mqsiprofile</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variables for WebSphere Message Broker.</td>
</tr>
<tr>
<td>mqsireportbroker</td>
<td>(broker name)</td>
<td>Mandatory</td>
<td>Gets the WebSphere Message Broker information.</td>
</tr>
<tr>
<td>mqsibrowse</td>
<td>(broker name) -t BROKERRESOURCES</td>
<td>Mandatory</td>
<td>Displays the Message Broker information resources.</td>
</tr>
<tr>
<td>echo</td>
<td>$ODBCINI</td>
<td>Optional</td>
<td>(On Unix only) This command is used only to create connections to IBM DB2. Prints strings in the output.</td>
</tr>
</tbody>
</table>

**IBM WebSphere Message Broker Flow (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mqsiprofile</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variables for WebSphere Message Broker.</td>
</tr>
<tr>
<td>mqsireportbroker</td>
<td>(broker name)</td>
<td>Mandatory</td>
<td>Gets the WebSphere Message Broker information.</td>
</tr>
<tr>
<td>ps</td>
<td>-ef</td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
<tr>
<td>mqsibrowse</td>
<td>(broker name) -t BROKERRESOURCES</td>
<td>Mandatory</td>
<td>Displays the Message Broker information resources.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>echo</td>
<td>$ODBCINI</td>
<td>Optional</td>
<td>(On Unix only) This command is used only to create connections to IBM DB2. Prints strings in the output.</td>
</tr>
<tr>
<td>db2</td>
<td>list database directory</td>
<td>Mandatory</td>
<td>Displays DB2 installation parameters.</td>
</tr>
<tr>
<td></td>
<td>and list node directory</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>IBM WebSphere Message Broker (on Unix)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mqsiprofile</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variables for WebSphere Message Broker.</td>
</tr>
<tr>
<td>mqsireportbroker</td>
<td>(broker name)</td>
<td>Mandatory</td>
<td>Gets the WebSphere Message Broker information.</td>
</tr>
<tr>
<td>mqsibrowse</td>
<td>(broker name) -t BROKERRESOURCES</td>
<td>Mandatory</td>
<td>Displays the Message Broker information resources.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only for creating the Web Services connections. Finds files and folders for the specific name.</td>
</tr>
<tr>
<td></td>
<td>-type</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>IBM WebSphere Message Broker (on Windows)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mqsiprofile</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variables for WebSphere Message Broker.</td>
</tr>
<tr>
<td>mqsireportbroker</td>
<td>(broker name)</td>
<td>Mandatory</td>
<td>Gets the WebSphere Message Broker information.</td>
</tr>
<tr>
<td>ps</td>
<td>-ef</td>
<td>Mandatory</td>
<td>Gets the process information.</td>
</tr>
<tr>
<td>mqsibrowse</td>
<td>(broker name) -t BROKERRESOURCES</td>
<td>Mandatory</td>
<td>Displays the Message Broker information resources.</td>
</tr>
</tbody>
</table>
### IBM WebSphere MQ (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dspmq</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the queue manager status.</td>
</tr>
<tr>
<td>dspmqver</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the queue manager version.</td>
</tr>
<tr>
<td>runmqsc</td>
<td>(queue manager)</td>
<td>Mandatory</td>
<td>Gets the queue information.</td>
</tr>
</tbody>
</table>

### IBM WebSphere MQ (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dspmq</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the queue manager status.</td>
</tr>
<tr>
<td>dspmqver</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the queue manager version.</td>
</tr>
<tr>
<td>runmqsc</td>
<td>(queue manager)</td>
<td>Mandatory</td>
<td>Gets the queue information.</td>
</tr>
</tbody>
</table>

### IBM WebSphere MQ (on z/OS)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>-a</td>
<td>grep &quot;+$port</td>
<td>Mandatory</td>
</tr>
<tr>
<td>df</td>
<td>-k</td>
<td>grep MQ`</td>
<td>Mandatory</td>
</tr>
<tr>
<td>echo</td>
<td>&quot;$netstat_info[1].mq_name +&quot;</td>
<td>cut -c1-4&quot;</td>
<td>Mandatory</td>
</tr>
<tr>
<td>oeconsol</td>
<td>&quot;$queue_manager&quot;</td>
<td>display qmgr ALL&quot;</td>
<td>Mandatory</td>
</tr>
<tr>
<td>oeconsol</td>
<td>&quot;$queue_manager&quot;</td>
<td>display queue (+</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>&quot;$entry_point.queue_name +&quot;) ALL</td>
<td>tr &quot;&quot;&quot;#&quot;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&quot;</td>
<td>tr &quot;&quot;&quot;&quot;&quot;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&quot;$name+&quot;&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>*oeconsol -*+ $queue_manager +* display channel(*) CONNAME where(QMNAME EQ &quot;$remote_queue_mngr_name +&quot;)&quot;</code></td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the channel information.</td>
</tr>
<tr>
<td><code>*oeconsol -*+ $queue_manager+&quot; display qlocal(*) &quot;</code></td>
<td></td>
<td>Mandatory</td>
<td>Gets the local queue information.</td>
</tr>
</tbody>
</table>

**IBM WebSphere MQ Queue (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dspmq</td>
<td></td>
<td>Mandatory</td>
<td>Gets the queue manager status.</td>
</tr>
<tr>
<td>dspmqver</td>
<td></td>
<td>Mandatory</td>
<td>Gets the queue manager version.</td>
</tr>
<tr>
<td>runmqsc</td>
<td>(queue manager)</td>
<td>Mandatory</td>
<td>Gets the queue information.</td>
</tr>
</tbody>
</table>

**IBM WebSphere MQ Queue (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dspmq</td>
<td></td>
<td>Mandatory</td>
<td>Gets the queue manager status.</td>
</tr>
<tr>
<td>dspmqver</td>
<td></td>
<td>Mandatory</td>
<td>Gets the queue manager version.</td>
</tr>
<tr>
<td>runmqsc</td>
<td>(queue manager)</td>
<td>Mandatory</td>
<td>Gets the queue information.</td>
</tr>
</tbody>
</table>

**Jboss Module (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists open ports.</td>
</tr>
</tbody>
</table>
### Jboss (on Unix and Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td>-name</td>
<td>Mandatory</td>
<td>Finds files and folders in the specified folder.</td>
</tr>
<tr>
<td>xargs</td>
<td>grep &lt;string&gt;</td>
<td>Optional</td>
<td>Finds strings in all files found in the output file.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### JRun (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

### JRun WAR Inc (on Unix and Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td>-name</td>
<td>Mandatory</td>
<td>Finds files and folders in the specified folder.</td>
</tr>
<tr>
<td>xargs</td>
<td>grep &lt;string&gt;</td>
<td>Optional</td>
<td>Finds strings in the files found in the output file.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### Microsoft Exchange CAS (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>tasklist</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists all running tasks.</td>
</tr>
<tr>
<td>del</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Deletes the XML file containing the old binding information.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>powershell</td>
<td>Add-PSSnapin Microsoft.Exchange.Management.PowerShell.E2010; Get-ExchangeServer -status -Identity (hostname)</td>
<td>Optional</td>
<td>This command is used only if you do not use TCP connections. Gets the CAS status, server roles, and Exchange setup structure. <strong>Caution:</strong> Do not use the dollar sign ($) in credentials for Exchange CAS, because Service Mapping uses the dollar sign for pattern variables in the powershell command.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only if PowerShell is not operational. Finds specific strings in the files and folders.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>
### Microsoft Dynamic CRM (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Microsoft BizTalk Orchestration (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
</tbody>
</table>

### Microsoft BizTalk Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>del</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Deletes the XML file containing the old binding information.</td>
</tr>
<tr>
<td>BTSTask</td>
<td>ExportBindings /</td>
<td>Optional</td>
<td>This command is used only if there are not MSSQL credentials. Extracts the binding information and places it into the XML file.</td>
</tr>
</tbody>
</table>
|         | Destination:%TEMP %
\ MyBindings.xml /  |                   |                                                  |
|         | Database:  |                   |                                                  |
|         | (MgmtDbName) /  |                   |                                                  |
|         | server:(serverName) |               |                                                  |

### Microsoft Exchange BackEnd Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>tasklist</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists all running tasks.</td>
</tr>
</tbody>
</table>

### Microsoft Exchange FrontEnd Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------</td>
<td>--------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>tasklist</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists all running tasks.</td>
</tr>
</tbody>
</table>

**Microsoft FAST Search Server (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

**Microsoft Exchange Hub Transport Server (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>tasklist</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists all running tasks.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>del</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Deletes the XML file containing the old binding information.</td>
</tr>
<tr>
<td>powershell</td>
<td>Add-PSSnapin Microsoft.Exchange.Management.PowerShell.E2010; Get-ExchangeServer -status -identity (entry_point.host_name)</td>
<td>Optional</td>
<td>This command is used only if you do not use TCP connections.</td>
</tr>
<tr>
<td></td>
<td>- export-clixml $env:TEMP \exchange_pwrshell_output.xml And</td>
<td></td>
<td>Gives the CAS status, server roles, and Exchange setup structure.</td>
</tr>
<tr>
<td></td>
<td>- export-clixml $env:TEMP \exchange_pwrshell_output.xml</td>
<td></td>
<td>Transport Server, because Service Mapping uses the dollar sign for pattern</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>variables in the powershell command.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Microsoft Internet Information Services (IIS) (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>appcmd.exe</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only for IIS version 7 and later. Archives information about the specified application name.</td>
</tr>
</tbody>
</table>

### Microsoft Internet Information Services (IIS) Virtual Directory (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>appcmd.exe</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only for IIS version 7 and later. Archives information about the specified application name.</td>
</tr>
<tr>
<td>iisapp.vbs</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only for IIS version 6. Archives information about the specified application name.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### Microsoft .NET commands (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ildasm.exe</td>
<td>text that contains process exe path</td>
<td>Mandatory</td>
<td>This file is necessary for performing the put file operation on the ILDisassembler alias. This file disassembles strings from the command .exe file.</td>
</tr>
<tr>
<td>.findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Gets the strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists all files in the specified folder.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>connection_strings_browser.exe</td>
<td>encrypted_configs[*].Name</td>
<td>Optional:</td>
<td>This command is used only for creating database encrypted connections. This file is necessary for performing the put file operation on the ConnectionStringsBrowser alias. Decrypts the database connections.</td>
</tr>
</tbody>
</table>

Microsoft Exchange Mailbox (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>tasklist</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists all running tasks.</td>
</tr>
<tr>
<td>del</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Deletes the XML file containing old binding information.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This command is used only if you do not use TCP connections. Gets the status, server roles, and the setup structure of Exchange Mailbox.

**Caution:** Do not use the dollar sign ($) in credentials for Exchange Mailbox, because Service Mapping uses the dollar sign for pattern variables in the powershell command.
<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only if PowerShell is not operational. Finds specific strings in the files and folders.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
</tbody>
</table>

**Microsoft SharePoint (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
</tbody>
</table>

**Microsoft SQL Database (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>sqlcmd</td>
<td>-Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nvcName +### +inboundtransporturl FROM adm_ReceiveloLocation rl, bts_receipeport rp where rl.receiveportid=rp.nid” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nID,nvcName,nApplicationID from bts_receipeport” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nID,nvcName,nApplicationID from bts_sendport” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nOrcPortID,nReceivePortID,nSendPortid from bts_orchestration_port_binding” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nID,nOrchestrationID from bts_orchestration_port” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nID,nvcFullName from bts_orchestration” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nID,nvcNamespace,bo.nvcName,ase.Name from bts_orchestration bo, adm_Host ah, adm_server ase, adm_Server2HostMapping ash where bo.nAdminHostID = ah.ID and ase.ID = ash.ServerID and ah.ID = ash.HostID” and -Stcp: (computer_system.primaryMan(entry_point.database)) -U (username) -P (password) -y 0 -h-1 -Q select nID,nSendPortID,nvcAddress from bts_sendport_transport”</td>
<td>Optional</td>
<td>This command is used only to discover Microsoft BizTalk. Gets the BizTalk information from the database.</td>
</tr>
</tbody>
</table>
**Microsoft Message Queueing (MSMQ) (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**Microsoft SQL Database (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>sqiservr.exe</td>
<td>Optional</td>
<td></td>
<td>Gets the version of the database from the executable file.</td>
</tr>
</tbody>
</table>

**MongoDB Database (on Windows or Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mongo</td>
<td>--eval db.printShardingStatus()</td>
<td>Mandatory</td>
<td>Retrieves the status of MondoDB that is used for load balancing.</td>
</tr>
<tr>
<td>mongo</td>
<td>--db.Name()</td>
<td>Mandatory</td>
<td>Retrieves the name of the database connected to MongoDB server.</td>
</tr>
</tbody>
</table>

**MySQL Server (on Windows and Linux)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings only from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>ps</td>
<td>--pid=(process.pid) --no-headers -o %U : %p : %a*</td>
<td>Optional</td>
<td>Gets the userid parameter value.</td>
</tr>
<tr>
<td>mysqld-nt/mysqld/mysqld.exe</td>
<td>-V</td>
<td>Mandatory</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>mysql</td>
<td>--user=(username) -- password=(password) -- port=(jdbc_port) -- skip-column-names -- silent --execute=&quot;SHOW ENGINE NDB STATUS;&quot; and --user=(username) -- password=(password) -- port=(jdbc_port) -- skip-column-names -- silent --execute=&quot;SHOW SLAVE STATUS;&quot; and --user=(username) -- password=(password) -- port=(jdbc_port) -- skip-column-names -- silent --execute=&quot;select host from information_schema.processlist where command like '% %binlog%%';&quot;</td>
<td>Optional</td>
<td>This command is used only to create cluster node connections, master replications, and slaves. Gets the deployment structure of the MySQL server.</td>
</tr>
<tr>
<td>which</td>
<td>mysql</td>
<td>Optional</td>
<td>This command is used only to create cluster node connections, master replications, and slaves. Gets the full path of MySQL cluster nodes.</td>
</tr>
</tbody>
</table>

MySQL Cluster MGM Node (on Linux or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>ndb_mgmd</td>
<td>-v and -e show</td>
<td>Mandatory</td>
<td>Gets the version and status of the MySQL Cluster MGM node.</td>
</tr>
</tbody>
</table>

Nginx (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nginx</td>
<td>-v</td>
<td>Mandatory</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>egrep</td>
<td>-v -e ^# -e &quot;#&quot;</td>
<td>Optional</td>
<td>This command is used if necessary to create HTTP connections. Ignores special characters.</td>
</tr>
</tbody>
</table>
### Oracle E-Business Suite (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>export</td>
<td>-</td>
<td>Mandatory</td>
<td>(For Unix only) Sets variables.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>sqlplus</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Creates the connection to the Oracle instance.</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Manipulates the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Gets open ports.</td>
</tr>
</tbody>
</table>

### Oracle Advanced Queue Queue (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>export</td>
<td>-</td>
<td>Mandatory</td>
<td>(On Unix only) Sets a variable.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
<tr>
<td>sqlplus</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Creates connection to the Oracle instance.</td>
</tr>
<tr>
<td>set</td>
<td>-</td>
<td>Optional</td>
<td>(On Windows only) Sets a variable.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists open ports.</td>
</tr>
</tbody>
</table>

### Oracle Concurrent Server (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists open ports.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Unix only) Manipulates the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**Oracle Discoverer Engine (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**Oracle Discoverer UI (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**Oracle Form UI (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**Oracle Fulfillment Server (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>
### Oracle HTTP Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Oracle Metric Client (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Oracle OACORE Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specified strings in the file and folders.</td>
</tr>
</tbody>
</table>

### Oracle OAFM Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specified strings in the file and folders.</td>
</tr>
</tbody>
</table>

### Oracle Database (on Unix)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>export</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variable.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>sqlplus</td>
<td>-s (username)/ (password)@(entry_point.instance)</td>
<td>Optional</td>
<td>This command is used only for getting the advanced queue information. Gets the Oracle version and information about advance queue.</td>
</tr>
</tbody>
</table>
| echo | -e set head off feed off pages 0 line 3000
\n\nselect NAME || QUEUE_TABLE || QID || RETENTION from DBA_QUEUES
where OWNER = UPPER((entry_point.scheme));
and
set head off feed off pages 0 line 3000
\n\nselect NAME || QUEUE_TABLE || QID || RETENTION from DBA_QUEUES
where OWNER = UPPER((entry_point.scheme));
and
select member as NEEBULA from v\$logfile;
and
select value as NEEBULA from v\$parameter where name='log_archive_dest';
and
select file_name as NEEBULA from dba_data_files;
and
select DB_LINK, HOST from DBA_DB_LINKS | Optional | This command is used only for getting the advance queue information. Gets the information about advance queue, storage devices, and database links. |
| lsnrctl | version
And
services | Optional | This command is used only for discovering the real application cluster (RAC) version. Gets the RAC version. |
### Oracle Database (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variable.</td>
</tr>
<tr>
<td>sqlplus</td>
<td>-s (username)/ (password)@[entry_point.instance]</td>
<td>Optional</td>
<td>This command is used only for getting the advance queue information. Gets the Oracle version and information about advance queue.</td>
</tr>
<tr>
<td>echo</td>
<td>-e set head off feed off pages 0 line 3000\n\n&quot;select NAME</td>
<td></td>
<td>'##'</td>
</tr>
<tr>
<td>lsnrctl</td>
<td>version</td>
<td>Optional</td>
<td>This command is used only for discovering the real application cluster (RAC) version. Gets the RAC version.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**Oracle iAS Web Module (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Optional</td>
<td>This command is used only to create the Web Services connections. Finds specified strings in the file and folders.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**Oracle Net Listener (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lsnrctl</td>
<td>version and status</td>
<td>Mandatory</td>
<td>Gets the version and status of the Oracle Net Listener.</td>
</tr>
<tr>
<td>export</td>
<td>-</td>
<td>Mandatory</td>
<td>Sets the variable.</td>
</tr>
<tr>
<td>crsctl</td>
<td>status res -t and config service -d</td>
<td>Mandatory</td>
<td>Gets the services status and configuration.</td>
</tr>
</tbody>
</table>

**Oracle PeopleSoft (on Unix)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>psadmin</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Retrieves information from the PeopleSoft repository or server.</td>
</tr>
<tr>
<td>tnsping</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Retrieves information about the database instance given to the tnsping command.</td>
</tr>
<tr>
<td>sqlplus</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Connects to the database instance and runs the sql query.</td>
</tr>
</tbody>
</table>
### Oracle Process Manager (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specified strings in the file and folders.</td>
</tr>
</tbody>
</table>

### Oracle Report Server (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specified strings in the file and folders.</td>
</tr>
</tbody>
</table>

### Oracle Tnslsnr Engine (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Oracle WebLogic Module (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

### Oracle WebLogic Server (version 10.3) (on Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>find</td>
<td>Various</td>
<td>Mandatory</td>
<td>Finds specific strings in files and folders.</td>
</tr>
</tbody>
</table>

**Oracle WebLogic On-demand Router Load Balancer (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>versionInfo.sh</td>
<td>-</td>
<td>Optional</td>
<td>Gets the Websphere Application Server version.</td>
</tr>
</tbody>
</table>

**Oracle Tuxedo (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output</td>
</tr>
<tr>
<td>dir</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Lists all existing files in the given folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only to create web service connections. Finds files and folders for specific name.</td>
</tr>
<tr>
<td>tmadmin</td>
<td>-v</td>
<td>Mandatory</td>
<td>Gets the Tuxedo version.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various</td>
<td>Mandatory</td>
<td>Gets open ports.</td>
</tr>
</tbody>
</table>

**Oracle Tuxedo Portal (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>dir</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Lists all existing files in the given folder.</td>
</tr>
<tr>
<td>find</td>
<td>Various</td>
<td>Mandatory</td>
<td>(For Windows only) Finds specific strings in files and folders.</td>
</tr>
<tr>
<td>find</td>
<td>-name</td>
<td>Optional</td>
<td>This command is used only to create web service connections. Finds files and folders for specific name.</td>
</tr>
</tbody>
</table>

**PostgreSQL Database (on Unix and Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>postgres</td>
<td>-v</td>
<td>Mandatory</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

**Red Hat Cluster Suite (on Linux)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ps</td>
<td>-ef</td>
<td>Mandatory</td>
<td>Displays the currently running process.</td>
</tr>
<tr>
<td>grep</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Searches plain-text data sets for lines that match the regular expression.</td>
</tr>
<tr>
<td>grep</td>
<td>-v &lt;string&gt;</td>
<td>Mandatory</td>
<td>Searches plain-text data sets for lines that do not match the regular expression.</td>
</tr>
<tr>
<td>ps</td>
<td>-ef</td>
<td>Mandatory</td>
<td>Checks if the corosync process is running on the server. If the process is running, the Red Hat Cluster is running on this server.</td>
</tr>
<tr>
<td>ifconfig</td>
<td>-</td>
<td>Mandatory</td>
<td>Displays the current network configuration information.</td>
</tr>
<tr>
<td>awk</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Manipulates the output.</td>
</tr>
<tr>
<td>hostname</td>
<td>-s</td>
<td>Mandatory</td>
<td>Displays the Fully Qualified Domain Name (FQDN).</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>hostname</td>
<td>-f</td>
<td>Mandatory</td>
<td>Displays the short host name.</td>
</tr>
</tbody>
</table>

**SAP BO BOXI ScheduleRouter (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>type</td>
<td>-</td>
<td>Mandatory</td>
<td>This command is used to create connections to Oracle.</td>
</tr>
<tr>
<td>tnsping</td>
<td>instance</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
</tbody>
</table>

**SAP Business Objects CMS Server (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>tnsping</td>
<td>instance</td>
<td>Optional</td>
<td>This command is used to create connections to Oracle.</td>
</tr>
</tbody>
</table>

**SAP Central Instance (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>disp + work</td>
<td>-V</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>ping</td>
<td>-n 1</td>
<td>Mandatory</td>
<td>(For Windows only) Verifies that the host is answering.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**SAP NetWeaver Dialog Instance (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>disp + work</td>
<td>-V</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>ping</td>
<td>-c 1</td>
<td>Mandatory</td>
<td>(For Unix only) Verifies that the host is answering.</td>
</tr>
<tr>
<td>ping</td>
<td>-n 1</td>
<td>Mandatory</td>
<td>(For Windows only) Verifies that the host is answering.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**SAP Evaluated Receipt Settlement (ERS) (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>disp + work</td>
<td>-V</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>ping</td>
<td>-n 1</td>
<td>Mandatory</td>
<td>(For Windows only) Verifies that the host is answering.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

**SAP HANA Database (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hdbsql</td>
<td>-v</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
</tbody>
</table>

**SAP Java Cluster (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>jlaunch</td>
<td>-V</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>ping</td>
<td>-n 1</td>
<td>Mandatory</td>
<td>(For Windows only) Verifies that the host is answering.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

**SAP Central Services (SCS) (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>msg_server</td>
<td>-V</td>
<td>Optional</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>ping</td>
<td>-n 1</td>
<td>Mandatory</td>
<td>(For Windows only) Verifies that the host is answering.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

**SQL Server Analysis Services (SSAS) (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsMdSrv.exe</td>
<td>-n</td>
<td>Mandatory</td>
<td>Gets the instance name.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

**SQL Server Integration Services (SSIS) Job (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsMdSrv.exe</td>
<td>-n</td>
<td>Mandatory</td>
<td>Gets the instance name.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specified strings in the file and folders.</td>
</tr>
<tr>
<td>sqlcmd</td>
<td>-Stcp: (computer_system.primaryManagementIP), (port) -y 0 -h-1 -E -Q&quot;select convert(varchar(max), convert(varbinary(max),packagedata)) from msdb.dbo.sysssispackages where name = '(name)' -o (tmp_dir)/(name).xml</td>
<td>Optional/ Mandatory</td>
<td>This command is used only for discovering jobs.</td>
</tr>
</tbody>
</table>

**SQL Server Integration Services (SSIS) (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>find</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Finds specified strings in the file and folders.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------------------------------------------------------</td>
<td>--------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>sqlcmd</td>
<td>-Stcp: (computer_system.primaryManagementIP) -h-1 -E -Q&quot;SELECT (name) + '###DB' FROM msdb.dbo.sysssispackages&quot; and -Stcp: (computer_system.primaryManagementIP) -h-1 -E -Q&quot;ELECT (name) + '###JOB' FROM msdb.dbo.sysjobs&quot;</td>
<td>Optional</td>
<td>This command is used only if you need to discover jobs.</td>
</tr>
</tbody>
</table>

**Sun Directory (on Solaris)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>slapd-dirserv</td>
<td>-D (instance) -v</td>
<td>Mandatory</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists the open ports.</td>
</tr>
</tbody>
</table>

**Sun JES (on Solaris)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>configutil</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the JES configuration.</td>
</tr>
</tbody>
</table>

**Sybase (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dataserverv</td>
<td>-v</td>
<td>Mandatory</td>
<td>Gets the version.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(For Windows only) Lists files and folders in the specified folder.</td>
</tr>
</tbody>
</table>

**Symantec Enterprise Vault (on Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>ping</td>
<td>-n 1[ip]</td>
<td>Mandatory</td>
<td>Verifies that the SQL server is running.</td>
</tr>
</tbody>
</table>

**TIBCO ActiveMatrix Adapter (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
</tbody>
</table>
### ServiceNow

#### New York

**IT Operations Management**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;IFS=$\n\n;for i in `find &quot; + $tra_home + &quot; -type f</td>
<td>grep -v /tmp</td>
<td>egrep \ .process$</td>
<td>$substvar$ &quot;; do result='$grep &quot; + Sentry_point.queue + &quot; $i&quot; echo $result; done</td>
</tr>
</tbody>
</table>

### TIBCO ActiveMatrix BusinessWorks (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TibcoFilesParser.ksh</td>
<td>-</td>
<td>Mandatory</td>
<td>(For Unix only) This file is necessary for performing the put file operation. It replaces all tags in Tibco configuration files and puts them under the /tmp folder.</td>
</tr>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>cd</td>
<td>-</td>
<td>Mandatory</td>
<td>(On Unix only) Changes directory.</td>
</tr>
<tr>
<td>chmod</td>
<td>-</td>
<td>Mandatory</td>
<td>(On Unix only) Changes file permissions.</td>
</tr>
<tr>
<td>for</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Runs loops.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Prints strings in the output.</td>
</tr>
<tr>
<td>if</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Starts a condition.</td>
</tr>
<tr>
<td>wc</td>
<td>-l</td>
<td>Mandatory</td>
<td>(On Unix only) Counts the output lines.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
</tbody>
</table>

### TIBCO ActiveMatrix BusinessWorks Process (on Unix or Windows)

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>-</td>
<td>Mandatory</td>
<td>Gets the hostname.</td>
</tr>
<tr>
<td>cd</td>
<td>-</td>
<td>Mandatory</td>
<td>(On Unix only) Changes directory.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>chmod</td>
<td>-</td>
<td>Mandatory</td>
<td>(On Unix only) Changes file permissions.</td>
</tr>
<tr>
<td>for</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Runs loops.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Prints strings in the output.</td>
</tr>
<tr>
<td>if</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Starts a condition.</td>
</tr>
<tr>
<td>wc</td>
<td>-l</td>
<td>Mandatory</td>
<td>(On Unix only) Counts the output lines.</td>
</tr>
<tr>
<td>dir</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Lists files and folders in the specified folder.</td>
</tr>
<tr>
<td>findstr</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Extracts strings from the output.</td>
</tr>
<tr>
<td>netstat</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Lists open ports.</td>
</tr>
<tr>
<td>cut</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Unix only) Splits the output line.</td>
</tr>
</tbody>
</table>

**TIBCO Enterprise Message Service (EMS) Queue (on Unix or Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rm</td>
<td>-f /tmp/ems.cmd</td>
<td>Mandatory</td>
<td>Prepares the EMS script.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>Prints strings in the output.</td>
</tr>
</tbody>
</table>

**TIBCO Enterprise Message Service (EMS) (on Unix and Windows)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rm</td>
<td>-f /tmp/ems.cmd</td>
<td>Mandatory</td>
<td>Prepares the EMS script.</td>
</tr>
<tr>
<td>echo</td>
<td>Various options</td>
<td>Mandatory</td>
<td>(On Windows only) Prints strings in the output.</td>
</tr>
</tbody>
</table>
SNMP-based queries
Service Mapping accesses network infrastructure devices like load balancers and routers using Simple Network Management Protocol (SNMP) v1/v2c/v3. Configure SNMP community credentials to enable this type of access.

Configure SNMP community if you use SNMP v1/v2 or SNMPv3 credentials if you use SNMP v3. For operational information, see SSH credentials.

SNMP-based queries have the format of strings of integers.

General SNMP-based queries

Make sure that the user has credentials for SNMPv3 unless read-only community string is used.

- 1.3.6.1.2.1.47.1.1.1.1.13
- 1.3.6.1.2.1.1.6
- 1.3.6.1.2.1.47.1.1.1
- 1.3.6.1.4.1.9.9.774.1.1.1
- 1.3.6.1.4.1.9.6

SNMP-based queries for switches and routers

- 1.3.6.1.4.1.9.9.46.1.3.1.1
- 1.3.6.1.4.1.9.9.68.1.2.1.1
- 1.3.6.1.2.1.17.1.4.1
- 1.3.6.1.2.1.17.4.3.1
- 1.3.6.1.2.1.17.7.1.2.2.1
- 1.3.6.1.2.1.17.2.15.1
- 1.3.6.1.4.1.9.9.23.1.2.1.1
- 1.3.6.1.4.1.45.1.6.13.2.1.1
- 1.0.8802.1.1.2.1.4.2.1
- 1.0.8802.1.1.2.1.4.1.1

A10 Thunder Series

- 1.3.6.1.4.1.1872.2.5.1.1.1.1
- 1.3.6.1.4.1.1872.2.5.3.1.3.2.1
- 1.3.6.1.4.1.1872.2.5.3.1.2.4.1
- 1.3.6.1.4.1.1872.2.5.3.1.3.1.10
- 1.3.6.1.4.1.1872.2.5.4.1.1.4.2.1
- 1.3.6.1.4.1.1872.2.5.3.1.6.3.1
- 1.3.6.1.4.1.22610.2.4.1.1.1
- 1.3.6.1.4.1.22610.2.4.1.6.2
- 1.3.6.1.4.1.22610.2.4.3.4.1.2.1
- 1.3.6.1.4.1.22610.2.4.3.2.1.2.1
- 1.3.6.1.4.1.22610.2.4.3.3.1.2.1
- 1.3.6.1.4.1.22610.2.4.3.4.3.1.1
- 1.3.6.1.4.1.22610.2.4.3.3.1.1
Big-Ip Local Traffic Manager (Ltm) F5 (On F5 Big-Ip) and Big-Ip Global Traffic Manager (Gtm) F5

- 1.3.6.1.4.1.3375.2.1.6.3
- 1.3.6.1.4.1.3375.2.1.4.1
- 1.3.6.1.4.1.3375.2.1.3.3.3
- 1.3.6.1.4.1.3375.2.1.6.2
- 1.3.6.1.4.1.3375.2.1.2.1.1.2.1
- 1.3.6.1.4.1.3375.2.2.10.1.2.1
- 1.3.6.1.4.1.3375.2.3.12.4.2.1
- 1.3.6.1.4.1.3375.2.3.12.1.2.1
- 1.3.6.1.4.1.3375.2.1.4.1
- 1.3.6.1.4.1.3375.2.1.6.2
- 1.3.6.1.4.1.3375.2.1.4.2
- 1.3.6.1.4.1.3375.2.2.10.1.2.1
- 1.3.6.1.4.1.3375.2.2.5.6.2.1
- 1.3.6.1.4.1.3375.2.2.10.8.2.1
- 1.3.6.1.4.1.3375.2.2.5.6.2.1
- 1.3.6.1.4.1.3375.2.2.10.1.2.1
- 1.3.6.1.4.1.3375.2.1.4.1
- 1.3.6.1.4.1.3375.2.1.4.2
- 1.3.6.1.4.1.3375.2.1.6.2
- 1.3.6.1.4.1.3375.2.3.11.1.2.1
- 1.3.6.1.4.1.3375.2.3.12.3.2.1
- 1.3.6.1.4.1.3375.2.3.12.4.2.1
- 1.3.6.1.4.1.3375.2.2.5.6.2.1
- 1.3.6.1.4.1.3375.2.3.12.3.2.1
- 1.3.6.1.4.1.3375.2.3.12.4.2.1
- 1.3.6.1.4.1.3375.2.3.12.5.2.1
- 1.3.6.1.4.1.3375.2.3.6.7.2.1
- 1.3.6.1.4.1.3375.2.3.11.4.2.1
- 1.3.6.1.4.1.3375.2.3.3.1.2.1
- 1.3.6.1.4.1.3375.2.3.3.3.2.1
- 1.3.6.1.4.1.3375.2.3.9.1.2.1
- 1.3.6.1.4.1.3375.2.3.9.3.2.1

Cisco Content Services Switch (CSS)

- 1.3.6.1.4.1.2467.1.16.4.1
- 1.3.6.1.4.1.2467.1.31.3
- 1.3.6.1.2.1.1.2
- 1.3.6.1.4.1.2467.4.1
- 1.3.6.1.4.1.2467.4.2
- 1.3.6.1.4.1.2467.4.3
- 1.3.6.1.4.1.2467.4.4
- 1.3.6.1.2.1.1.1
- 1.3.6.1.4.1.2467.1.15.2.1
- 1.3.6.1.4.1.2467.1.16.4.1
Cisco Global Site Selector (GSS)

- 1.3.6.1.2.1.1.5 – to get the hostname.
- 1.3.6.1.2.1.1.1 – to get product name.

Citrix Netscaler

- 1.3.6.1.4.1.5951.4.1.3.6.1
- 1.3.6.1.4.1.5951.4.1.3.2.1
- 1.3.6.1.4.1.5951.4.1.3.1.1
- 1.3.6.1.4.1.5951.4.1.2.7.1
- 1.3.6.1.4.1.5951.4.1.2.2.1
- 1.3.6.1.4.1.5951.4.1.2.1.1
- 1.3.6.1.4.1.5951.4.1.1.6
- 1.3.6.1.4.1.5951.4.1.1.26.1
- 1.3.6.1.4.1.5951.4.1.1.24.1
- 1.3.6.1.4.1.5951.4.1.1.14
- 1.3.6.1.2.1.55.1.8.1
- 1.3.6.1.2.1.47.1.1.1
- 1.3.6.1.2.1.4.20.1
- 1.3.6.1.2.1.2.2.1
- 1.3.6.1.2.1.2.2.1
- 1.3.6.1.2.1.1.5

Citrix Netscaler Global Server Load Balancer

- 1.3.6.1.4.1.5951.4.1.1.51.3.1.1
- 1.3.6.1.4.1.5951.4.1.2.1.1

IBM WebSphere DataPower

- 1.3.6.1.2.1.1.1
- 1.3.6.1.2.1.1.5
- 1.3.6.1.4.1.14685.3.1.11.1

Juniper Junos

- 1.3.6.1.4.1.2636.3.1.3
**Network Appliance Filer Storage Array**

- 1.3.6.1.4.1.789.1.1.2
- 1.3.6.1.4.1.789.1.1.9

**Radware Alteon NG**

- 1.3.6.1.4.1.1872.2.5.4.2.32.1
- 1.3.6.1.4.1.1872.2.5.1.1.1.1
- 1.3.6.1.4.1.1872.2.5.3.1.3.2.1
- 1.3.6.1.4.1.1872.2.5.3.1.2.4.1
- 1.3.6.1.4.1.1872.2.5.1.3.1.10
- 1.3.6.1.4.1.1872.2.5.4.1.1.4.2.1
- 1.3.6.1.4.1.1872.2.5.3.1.6.3.1
- 1.3.6.1.4.1.1872.2.5.4.1.1.4.2.1
- 1.3.6.1.2.1.1.1
- 1.3.6.1.4.1.1872.2.5.4.1.1.4.5.1
- 1.3.6.1.4.1.1872.2.5.4.1.1.3.3.1
- 1.3.6.1.4.1.1872.2.5.4.1.1.2.2.1
- 1.3.6.1.2.1.4.24.4.1
- 1.3.6.1.4.1.89.35.1.13.1.2
- 1.3.6.1.4.1.89.2.12
- 1.3.6.1.4.1.89.2.4
- 1.3.6.1.4.1.89.35.1.40.52.1.1
- 1.3.6.1.4.1.89.35.1.13.1
- 1.3.6.1.4.1.89.35.1.30.1
- 1.3.6.1.2.1.68.1.3.1
- 1.3.6.1.4.1.89.35.1.13.1
- 1.3.6.1.4.1.89.35.1.11.1
- 1.3.6.1.4.1.89.35.1.40.52.1.1

**Create a discovery behavior for Unix-based load balancers**

If your network uses load balancers running on Linux hosts, create a discovery behavior to ensure that Service Mapping and Discovery discover them correctly.

Make sure that Discovery is activated.

Role required: sm_admin or admin

The discovery and mapping process begins with Discovery identifying the host. After the host discovery is complete, Service Mapping starts discovering and mapping applications running on this host.

The default discovery process identifies a configuration item (CI) as a Linux server if it finds an SSH port in use. It may lead to incorrect discovery of load balancers running on Linux as Linux servers. Create a discovery behavior to first discover network devices using SNMP protocol. After Discovery has discovered Unix-based load balancers correctly and created the appropriate CIs for them, you can continue with discovering and mapping application services in Service Mapping.

Discovery performs discovery through a MID Server located in the enterprise private network. A behavior determines:

- What MID Server Discovery is using.
1. Create a new behavior for discovering Unix-based load balancers:
   a) Navigate to Service Mapping > Administration > Behavior.
   b) Click New.
   c) Enter the name for this behavior.
   d) Right-click on the header bar and click Save.
   e) Click New on the Discovery Functionality tab.
   f) Enter 1 in the Phase field.
   g) Select SNMP only from the Functionality definition list.
   h) Select the relevant MID Servers in the MID Servers area.
   i) Check the Active check box.
   j) Click Submit.

2. Create a discovery schedule for the newly defined behavior:
   a) Navigate to Discovery > Discovery Schedules.
   b) Click New.
   c) Fill in the fields as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique, descriptive name for your schedule.</td>
</tr>
<tr>
<td>Discover</td>
<td>Select Configuration items.</td>
</tr>
<tr>
<td>MID Server selection method</td>
<td>Select Use behavior.</td>
</tr>
<tr>
<td>Behavior</td>
<td>Select the behavior you created in step 1.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to enable this schedule. If you clear the check box,</td>
</tr>
<tr>
<td></td>
<td>the schedule is disabled, but you can still run a discovery manually from</td>
</tr>
<tr>
<td></td>
<td>this form, using the configured values.</td>
</tr>
<tr>
<td>Max run time</td>
<td>Set a time limit for running this schedule. When the configured time</td>
</tr>
<tr>
<td></td>
<td>elapses, the remaining tasks for the discovery are canceled, even if the</td>
</tr>
<tr>
<td></td>
<td>scan is not complete. Use this field to limit system load to a desirable</td>
</tr>
<tr>
<td></td>
<td>time window. If no value is entered in this field, this schedule runs until</td>
</tr>
<tr>
<td></td>
<td>complete.</td>
</tr>
<tr>
<td>Run and related fields</td>
<td>Determines the run schedule of the discovery. Configure the frequency in</td>
</tr>
<tr>
<td></td>
<td>the Run field and the other fields that appear to specify an exact time.</td>
</tr>
</tbody>
</table>

   Note: The run time always uses the system time zone. If you add the optional Run as tz field, it has no effect on the actual runtime.

3. Create an IP range for the discovery schedule:
The IP range for the schedule must correlate with the IP range of the MID Server that you assigned to the behavior. The MID Server must also have the appropriate application or the ALL application.

a) On the Discovery IP Ranges tab, click New.
b) Enter the name for this IP range.
c) Configure the IP range by entering IP addresses in the Starting IP and Ending IP fields.

4. Click Submit.

Verify that Service Mapping is set up properly

Before you start mapping application services, verify that you configured MID Servers to discover application services and provided all the necessary host and application credentials.

Role required: sm_admin

Service Mapping performs an automatic scan of the following resources and configurations to give you the readiness status:

- (Mandatory) There are MID Servers with correctly configured IP ranges and the applications and capabilities settings. For more information, see MID Server configuration for Service Mapping.
  
  If this check does not recognize correctly configured MID Servers, you cannot perform the bulk discovery.
- (Optional) There are credentials for operating systems and applications as described in Prerequisites for performing top-down discovery using Service Mapping.
- (Optional) There are at least three load balancers recently discovered during horizontal discovery.
- (Optional) There are at least 100 hosts discovered by the horizontal discovery in the last 14 days.
- (Optional) The data collection using Netflow or VPC Flow Logs is configured. This configuration is necessary to enhance the traffic-based discovery in Service Mapping. For more information, see Data collection and discovery using Netflow and Data collection and discovery using VPC Flow Logs.
- (Optional) Cloud Discovery is activated and configured. This setting is necessary to discover application services based on IaaS.

If an optional resource or configuration is missing, you can still map application services, but it may result in partial discovery of some CIs and connections.

1. Navigate to Service Mapping > Home.
   The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.

2. Verify that Service Mapping is set up and ready for discovery.
   If some obligatory configurations are missing, the error displays at the top of the window.
   a) Click the link in the error or click the Readiness Checklist icon.
b) In the Service Mapping Readiness Checklist window, click the link next to the setup category with the error icon (✓). You cannot perform discovery and mapping unless there are no errors in the Readiness Checklist window.

c) Perform the necessary action to complete the setup. For example, configure missing credentials.

3. If more than one setup category has an error, return to the Readiness Checklist window and fix any other errors.

Map business services in bulk

Quick start tests for Service Mapping

Validate that Service Mapping still works after you make any configuration change such as apply an upgrade or develop an application. Copy and customize these quick start tests to pass when using your instance-specific data.

Service Mapping quick start tests require activating the Service Mapping plugin (com.snc.service-mapping) and loading the demo data.

<table>
<thead>
<tr>
<th>Test</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SM OOTB: Tests SM application visibility by roles</td>
<td>Validate the visibility of Service Mapping applications and modules for different roles. For example, the test verifies that a user logged in with the sm_user role, cannot access the Administration module under Service Mapping.</td>
</tr>
</tbody>
</table>
**Application service mapping**

You can perform initial mapping of multiple or single application services in your organization using a method that best suits you.

You can map application services after you complete the Service Mapping setup.

---

**Service Mapping workflow**
A typical organization has hundreds of application services. Mapping each of them individually can be time and effort consuming. With Service Mapping, you can map most IT services in bulk. Mapping multiple application services at once is very effective, but is not precise. You may need to individually map application services omitted during bulk mapping.

A discovery method depends on how you have been managing your application services prior to mapping them and what information about them you gathered:

- **Map multiple application services suggested by Service Mapping**

  Service Mapping uses load balancers on your network to identify possible application services (candidates). You can map all or selected candidate in bulk, creating multiple application services in a single operation. This method suits your organization if you do not have much information about application services.

- **Map multiple application services from a CSV file**

  This method suits you if your organization has performed cross-organization mapping and analysis and collected some information about planned application services. If so, you can organize the collected information in a specific order and save it as a CSV file. Service Mapping extracts information from this file and creates potential application services referred to as service candidates.

- **Map a single application service**

  You can map individual, single application services one by one by defining attributes for each application service. Use this mapping method for application services Service Mapping cannot discover during bulk mapping. You must know or plan to find out application service attributes necessary for mapping.

**Map multiple application services suggested by Service Mapping**

You can map multiple application services identified and suggested by Service Mapping in a single operation. This method suits your organization if you do not have much information about business services.

Role required: sm_admin

You must have Service Mapping enabled to map or review application services.

Immediately after you complete the Service Mapping setup, Service Mapping automatically creates potential application services referred to as candidates:

1. Service Mapping extracts entries directly from load balancers on your network.
2. Service Mapping converts load balancer entries into potential entry points.
   - It also scans tables containing traffic-based connections to identify potential entry points. An entry point is how clients access an application service. Service Mapping starts the mapping process from this point. For example, to map your electronic mailing application service, define an IP address or host name of the email server as an entry point.
3. Service Mapping checks that potential entry points created from traffic-based connections are unique and not in use by any existing application services to avoid duplication.
4. Service Mapping creates an application service candidate for each entry point.
   - Not all candidate application services correlate to actual application services. Service Mapping filters out unlikely candidates by applying an algorithm that takes into account such parameters as empty IP addresses, port numbers, and the load balancer VIP hit counts less than 1000. Service Mapping stores candidates in the Candidate Entry Point (sa_cand_entry_point) table in the CMDB.

During the discovery and mapping process, Service Mapping creates an application service from each candidate and assigns it the entry point retrieved from the candidate. While mapping in
bulk is an effective and fast way to create numerous application services, the result of the bulk mapping is not precise. There may be some false or duplicate application services created by mistake because the data from the load balancer is raw. During the review and approval process, you review all mapped application services and get rid of the ones that were erroneously created.

You can further filter the list of candidates by selecting specific ones to include in or exclude from discovery. Mapping only selected candidates provides the following advantages:

- Reduces the discovery time
- Minimizes the number of potential irrelevant application services

You can remove the unwanted candidates from the list using these methods either by filtering them out or deleting them from the Candidate Entry Point (sa_cand_entry_point) table. In both cases, you can bring hidden or deleted candidates back to the list.

1. Navigate to Service Mapping > Home.
   The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.

2. Verify that Service Mapping is set up and ready for discovery.
   If some obligatory configurations are missing, the error displays at the top of the window.
   a) Click the link in the error or click the Readiness Checklist Icon.
   b) In the Service Mapping Readiness Checklist window, click the link next to the setup category with the error icon (❌).
      You cannot perform discovery and mapping unless there are no errors in the Readiness Checklist window.
   c) Perform the necessary action to complete the setup.
      For example, configure missing credentials.
3. To map all application services suggested by Service Mapping, click **Map**.

4. To map selected application services suggested by Service Mapping:
   a) Click **Additional options** under the **Map** tile.
   b) Select **View Candidates List**.
   c) Optional: Display candidates based on traffic-based connections, remove the condition that filters out candidates retrieved from load balancers or imported from a CSV file.

   ![Candidate Entry Points](image)

   The candidate list shows candidates with the source attribute **netstat**.

d) Review the list of candidates to decide which ones you want to discover as application services.

e) Create application services from the relevant candidates:
If you want to create application services from most of the candidates

1. Filter out irrelevant candidates by selecting them and clicking **Actions on selected rows**, and then clicking **Ignore Selected**.

   The irrelevant candidates are hidden from the list.

   **Note:** To show candidates you previously filtered out, set the **Ignore** filtering condition to **true**.

2. Alternatively, delete irrelevant candidates by selecting them and clicking **Actions on selected rows**, and then selecting **Delete**.

   The irrelevant candidates are removed from the Candidate Entry Point (sa_cand_entry_point) table containing candidates.

   **Note:** Service Mapping recreates deleted candidates every time Discovery performs horizontal discovery of the same load balancers or you reimport candidates from the same CSV file.

3. Click **Discover All**.

If you want to create application services only from several candidates

1. Select the check boxes next to the relevant candidates.

2. Click **Actions on selected rows** at the bottom of the screen.

3. Click **Discover selected**.

Service Mapping starts discovery of all candidates and creates application services from them. If you have many candidates, the discovery may take some time.

After the discovery process finishes, application services, which Service Mapping discovered without errors and with at least one discovered CI, appear under the **Approve** tile. Application services discovered with errors appear under the **Fix** tile.

- Review the list of created application services. If Service Mapping did not create some application services during bulk mapping as you expected, create them manually as described in **Map a single application service**.
- Fix application services discovered with errors. For more information, see **Fix application service errors in bulk**.
- Send application services discovered without errors to their respective owners for approval. For more information, see **Send application service maps for review**.
Map multiple application services from a CSV file

This method suits you if your organization has performed cross-organization mapping and analysis and collected some information about planned application services. If so, you can organize the collected information in a specific order and save it as a CSV file. Service Mapping extracts information from this file and creates potential application services referred to as service candidates.

Perform the procedure described in Prepare a CSV file for mapping your candidates.

⚠️ Warning: If you import planned services using the Service Map Planner module, use the CSV file format described in Import planned application services from the CSV file.

Role required: sm_admin

You must have Service Mapping enabled to map or review application services.

If necessary, you can import service candidates from multiple CSV files.

When reimporting application service candidates, Service Mapping does not import candidates you chose to ignore earlier.

After Service Mapping discovers configuration items (CIs) belonging to your application service for the first time, it then rediscovers CIs to find changes and updates. Create or modify discovery schedules to control how often Service Mapping rediscovers services or CIs.

1. Navigate to Service Mapping > Home.
   The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.
2. Click Additional Options under the Map tile.
3. Click Import Service Map List.
4. In the Import CSV file window, click Choose file.
5. Navigate to the CSV file to use for the import and click Choose.
6. Click Import.
   The imported application service candidates are added to the list of candidates.
7. Check that the overall number of service candidates on the Map tile increased by the expected number.
8. Click Map to create application services from the candidates you imported from the CSV file.

   • Review the list of created application services. If Service Mapping did not create some application services during bulk mapping as you expected, create them manually as described in Map a single application service.
   • Fix application services discovered with errors. For more information, see Fix application service errors in bulk.
   • Send application services discovered without errors to their respective owners for approval. For more information, see Send application service maps for review.

Prepare a CSV file for mapping your candidates

Organize information about potential application services (candidates) in your organization and save it in a CSV file.

Role required: sm_admin
**Warning:** If you import planned services using the Service Map Planner module, use the CSV file format described in *Import planned application services from the CSV file*.

2. Define entry point attributes for service candidates in the .xlsx file. Customers use these entry points to access services. For example, they use http://www.google.com:8080 to access the Google page.

**Warning:** Do not confuse column order in the .xlsx file. Entering attributes in wrong columns causes mapping errors.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Enter a unique name of the planned application service. For example, NY Public Library or Google. <strong>Warning:</strong> Service Mapping imports only application services with unique names. If there is more than one entry with the same name in CSV files you use, the latest imported record overwrites the previous ones.</td>
</tr>
<tr>
<td>B</td>
<td>(Optional) If you know the URL that serves as the entry point to your application service, enter it in this column. For example, enter <a href="https://www.nypl.org/">https://www.nypl.org/</a> or <a href="http://www.google.com:8080">www.google.com:8080</a>. <strong>Warning:</strong> If you define the URL, you do not have to define the IP address, port, FQDN, and protocol. Service Mapping parses the URL to extract the IP address, port, and FQDN. If there is no URL that serves as the entry point, define other parameters in columns C, and E.</td>
</tr>
<tr>
<td>C</td>
<td>(Optional) The IP address of the entry point in the IPv4 format. Enter this parameter only if you did not enter the URL in column B.</td>
</tr>
<tr>
<td>D</td>
<td>(Optional) The port of the entry point. For example, 8080 for the Google page. Enter this parameter only if you did not enter URL in column B. If you did not enter either URL or the port, Service Mapping uses port 80 for the http protocol and port 443 for the https protocol.</td>
</tr>
<tr>
<td>E</td>
<td>(Optional) The FQDN of the service entry point, for example, <a href="http://www.google.com">www.google.com</a> for Google. Enter this parameter only if you did not enter the URL in column B.</td>
</tr>
</tbody>
</table>
3. Save the file in a drive that you can access during the import with the .csv extension.

Map multiple application services from a CSV file

Map a single application service

In addition to mapping application services in bulk, you can map individual, single application services by defining attributes for each application service. Use this mapping method if you already know or are planning to find out application service details. You must have Service Mapping enabled to map or review application services.

- Verify that Service Mapping is set up properly.
- If the application service relies on PaaS supported by Microsoft Azure or Amazon Web Services, [discover datacenters] and then create [horizontal discovery schedules] on the cloud service account.

Role required: sm_admin

You can map single application services in addition to bulk mapping, which is very effective at mapping multiple application services, but is not precise. You may need to individually map application services omitted during bulk mapping.

The most important attribute you must know and configure for your application service is an entry point.

An entry point is how clients access an application service. Usually, it is either a URL or a combination of the IP address and port. Service Mapping starts the mapping process from this point. For example, to map your electronic mailing application service, define an IP address or host name of the email server as an entry point.

Entry points vary depending on the nature of the application service. Service Mapping comes with a wide range of preconfigured entry point types that cover many commonly used applications.

In addition to entry points that Service Mapping can discover, you may add entry points that Service Mapping cannot discover. You may want to add such an entry point as an indication that your application service contains a device or application.

Also, you must define an owner for your application service. The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes.

After Service Mapping discovers configuration items (CIs) belonging to your application service for the first time, it then rediscovers CIs to find changes and updates. Create or modify discovery schedules to control how often Service Mapping rediscovers services or CIs.

All application services, both manually created and discovered by Service Mapping, are stored in the Application Service [cmdb_ci_service_discovered] table.

1. Navigate to Service Mapping > Home.
   The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.
2. Click Additional options under the Map file.
3. Click Define A Single Service Map.
4. Define the basic application service attributes:
### 5. If you do not know the entry points, request them from the application service owner:

   a) Click the **Request data** link.
   
   The Request for data window opens.

   b) If the **Assigned to** list is blank, select the name of the application service owner.

   c) Enter your questions about this application service in the **Description** field.
   
   For example, you can ask what the entry points are.

   d) Click **Submit**.
   
   The system opens ServiceNow task for the owner and automatically sends an email notification.

   e) Once the owner provides missing information, you receive an email notification. Click the link in the email to open the Questionnaire screen for the relevant application service.

   f) Review information added by the owner on the **Other Entry Points** and **Components** tabs.

   g) Continue with the next step.

### 6. To add an entry point that Service Mapping can discover, perform these steps.

   a) Click **Discoverable by Service Mapping**.
   
   The page offers tiles for applications most commonly used for entry points.

   b) Click the tile matching the application to which the entry point belongs. For example, **Web Application** if you create an application service with a web interface. If tiles do not contain the necessary application, click **See More Types** to choose from a list of all preconfigured entry point types.

   
   
   **Note:**

   If the required entry point type does not appear in the list, you may need to create it as described in [Create entry point types for Service Mapping](#).

   c) Configure attributes for the entry point as described in **Entry point attributes**.
   
   Entry point parameters depend on the type you select.

   d) Click **Add**.
   
   The entry point is added under **Entry Points**.

   e) To add more entry points, click **Add Entry Point** and repeat step 6.

### 7. To add a static entry point that Service Mapping cannot discover, perform these steps.

   a) Click **Manually created**.
b) Define the CI to serve as the entry point.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Select the CI type (CI class) for the CI you are adding. Every CI belongs to a CI type which contains a set of attributes configured for this kind of CI, for example, <code>cmdb_ci_appl</code> for applications.</td>
</tr>
<tr>
<td>CI Name</td>
<td>Select the CI from the list of CIs of the selected CI type.</td>
</tr>
</tbody>
</table>

The CI type list includes only allowed CI types. For example, you cannot add an application cluster.

c) Click **Submit**.

8. To save attributes you have defined up until this moment and carry on configuring attributes for this application service, click **Save** or **Update**.
Service Mapping saves the new application service in the application service (`cmdb_ci_service_discovered`) table. It also starts the discovery process in the background.

9. To save definitions, discover this application service, and navigate to the map for this application service, click **Discover**.
Service Mapping starts the discovery process. The map for this application service page opens and displays the mapping in progress. The application service status changes to In Progress.

- If Service Mapping discovers the application service with errors, see [fix errors in individual application service maps](#).
- Send application services discovered without errors to their respective owners for approval. For more information, see [Send application service maps for review](#).

**Entry point attributes**

Check attribute definitions to correctly add entry points to your application services. Service Mapping comes with a wide range of preconfigured entry point types that cover many commonly used applications.

An entry point is how clients access an application service. Entry points are stored as records in the Endpoint (`cmdb_ci_endpoint`) tables.

There are some general attributes shared by almost all entry point types as described in the following table:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host/Hostname/Host Name</td>
<td>The value of the target server on which the service is running. This value can contain a real host name, alias, IP, or VIP.</td>
</tr>
<tr>
<td>Port</td>
<td>The port number of the service that you want to discover.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the service that you want to discover.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>URL</td>
<td>The URL of the service that you want to discover.</td>
</tr>
</tbody>
</table>

There are some attributes which you configure differently depending on what entry point they relate to, as described in the following table:

**Entry point attributes specific for some entry points**

<table>
<thead>
<tr>
<th>Entry point type</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Directory Domain to Domain Controllers Endpoint</td>
<td>Domain name</td>
<td>The AD domain name.</td>
</tr>
<tr>
<td></td>
<td>Forest name</td>
<td>The AD forest name.</td>
</tr>
<tr>
<td></td>
<td>Domain controller distinguished name</td>
<td>The distinguished name of the AD domain controller. Usually it is a compilation of the domain controller name and the hostname.</td>
</tr>
<tr>
<td></td>
<td>Domain Controller Name</td>
<td>The AD Domain Controller name.</td>
</tr>
<tr>
<td>Active Directory Forest Endpoint</td>
<td>Forest name</td>
<td>The AD Forest name.</td>
</tr>
<tr>
<td>AWS Resource Name Endpoint</td>
<td>ARN</td>
<td>The Amazon Resource name.</td>
</tr>
<tr>
<td>Application to Storage Endpoint</td>
<td>File in use by application</td>
<td>The path to the file used by an application. For example, mssql uses c:\applications\mssql\users.db. Service Mapping populates this attribute using patterns.</td>
</tr>
<tr>
<td></td>
<td>File system name</td>
<td>The file system containing the file used by an application. For example, if a mssql uses c:\applications\mssql\users.db, the value is &quot;c:&quot;.</td>
</tr>
<tr>
<td></td>
<td>Connected array ids</td>
<td>N/A</td>
</tr>
<tr>
<td>Advanced Queue</td>
<td>Scheme</td>
<td>The name of the Oracle DB table where the Queue is defined.</td>
</tr>
<tr>
<td></td>
<td>Instance</td>
<td>The name of the Oracle instance.</td>
</tr>
<tr>
<td></td>
<td>Queue</td>
<td>The name of the queue.</td>
</tr>
<tr>
<td>BizTalk Connection</td>
<td>Adapter address</td>
<td>The connect string to the Biztalk adapter. It can be a path, URL, host, or port.</td>
</tr>
<tr>
<td></td>
<td>Transport Type</td>
<td>The type of the Biztalk adapter. For example a URL, file SOAP, or FTP.</td>
</tr>
<tr>
<td>CRM Component</td>
<td>Role</td>
<td>The role of the CRM. Leave empty if unknown.</td>
</tr>
<tr>
<td>DB2 UDB</td>
<td>Instance</td>
<td>The name of the DB2 instance.</td>
</tr>
<tr>
<td>DCTM Connection to broker</td>
<td>Repository</td>
<td>The name of documentation repository.</td>
</tr>
<tr>
<td>Entry point type</td>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DCTM Index Agent Connection</td>
<td>Instance Name</td>
<td>The name of the instance of Documentation Index Agent.</td>
</tr>
<tr>
<td></td>
<td>Repository</td>
<td>The name of the documentation repository.</td>
</tr>
<tr>
<td>DCTM Job Processor</td>
<td>Repository</td>
<td>The name of documentation repository.</td>
</tr>
<tr>
<td>Citrix Delivery Controller</td>
<td>Farm Type</td>
<td>The Type of the Delivery Controller.</td>
</tr>
<tr>
<td></td>
<td>Controller Name</td>
<td>The name of the Citrix Controller.</td>
</tr>
<tr>
<td>DocBase Connection</td>
<td>Repository</td>
<td>The name of the documentation repository.</td>
</tr>
<tr>
<td>EJB</td>
<td>JNDI Name</td>
<td>The jndi name of target EJB.</td>
</tr>
<tr>
<td>EMS</td>
<td>Queue</td>
<td>The name of Tibco EMS queue.</td>
</tr>
<tr>
<td>EMS JNDI</td>
<td>JNDI Name</td>
<td>The jndi name of Tibco EMS connection.</td>
</tr>
<tr>
<td>F5 Mirror</td>
<td>Origin IP</td>
<td>The IP address of the second pair F5 device.</td>
</tr>
<tr>
<td></td>
<td>Origin host</td>
<td>The hostname of the second pair F5 device.</td>
</tr>
<tr>
<td>Fast Search</td>
<td>Repository</td>
<td>The name of repository of fast search.</td>
</tr>
<tr>
<td>ITAM Asset Center</td>
<td>Folder</td>
<td>The path to configuration folder.</td>
</tr>
<tr>
<td>JMS Flow</td>
<td>JMS project</td>
<td>The name of Tibco project.</td>
</tr>
<tr>
<td>JMS Flow</td>
<td>Queue Name</td>
<td>The name of Tibco EMS queue.</td>
</tr>
<tr>
<td>JMS Server</td>
<td>Queue Name</td>
<td>The name of the queue defined on JMS server.</td>
</tr>
<tr>
<td>HTTP Listener to WMB Dependency</td>
<td>Directory</td>
<td>The path to the WMB installation directory.</td>
</tr>
<tr>
<td>LDAP</td>
<td>Protocol</td>
<td>The protocol name: ldap.</td>
</tr>
<tr>
<td></td>
<td>Type</td>
<td>The LDAP query type. This attribute is not supported. Leave empty.</td>
</tr>
<tr>
<td>MQ</td>
<td>Queue Name</td>
<td>The name of the IBM MQ Queue.</td>
</tr>
<tr>
<td>MQ Flow</td>
<td>Source IP</td>
<td>The IP of the source IP create connection to MQ.</td>
</tr>
<tr>
<td></td>
<td>Queue Name</td>
<td>The name of IBM MQ Queue.</td>
</tr>
<tr>
<td>MQ Cross-memory</td>
<td>Command line</td>
<td>The command line of MQ.</td>
</tr>
<tr>
<td></td>
<td>Queue Name</td>
<td>The name of IBM MQ Queue.</td>
</tr>
<tr>
<td>MSMQ Flow</td>
<td>Queue Name</td>
<td>The name of MS MQ queue.</td>
</tr>
<tr>
<td>MS SQL Server</td>
<td>Instance</td>
<td>The name of MSSQL instance.</td>
</tr>
<tr>
<td>MySQL Cluster Data Node</td>
<td>Node ID</td>
<td>The ID of MySQL instance.</td>
</tr>
<tr>
<td></td>
<td>Node group</td>
<td>The group of MySQL instance.</td>
</tr>
<tr>
<td>Entry point type</td>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>MySQL Cluster MGM</td>
<td>Node ID</td>
<td>The ID of MySQL instance.</td>
</tr>
<tr>
<td>MySQL Slave Server</td>
<td>Config IP</td>
<td>The IP of the MySQL Config server.</td>
</tr>
<tr>
<td></td>
<td>Config Port</td>
<td>The port of the MySQL Config server.</td>
</tr>
<tr>
<td>Oracle DB</td>
<td>Schema</td>
<td>The name of table on the Oracle instance.</td>
</tr>
<tr>
<td></td>
<td>Instance</td>
<td>The name of oracle instance.</td>
</tr>
<tr>
<td>Oracle ESB Connection</td>
<td>Adapter data</td>
<td>The content of the adapter.</td>
</tr>
<tr>
<td></td>
<td>Adapter Type</td>
<td>The type of the adapter.</td>
</tr>
<tr>
<td>Oracle RAC DB</td>
<td>Service Name</td>
<td>The name of the Oracle RAC service.</td>
</tr>
<tr>
<td>PostgreSQL DB</td>
<td>Instance</td>
<td>The name of the PostgreSQL DB.</td>
</tr>
<tr>
<td>Remote TCP Cross-memory</td>
<td>Source IP</td>
<td>The IP of the source host process.</td>
</tr>
<tr>
<td></td>
<td>Source Port</td>
<td>The port of source process.</td>
</tr>
<tr>
<td>SAP APP EP</td>
<td>SID</td>
<td>The SAP instance SID name.</td>
</tr>
<tr>
<td></td>
<td>Instance</td>
<td>The instance name.</td>
</tr>
<tr>
<td>SAP BO Servers</td>
<td>Server name</td>
<td>The name of the BO server.</td>
</tr>
<tr>
<td>Sharepoint connection</td>
<td>Origin URL</td>
<td>The URL of the IIS.</td>
</tr>
<tr>
<td>SSAS</td>
<td>Project</td>
<td>The name of the SSAS project (can be empty).</td>
</tr>
<tr>
<td>SSAS for MSSQL</td>
<td>Instance</td>
<td>The name of the MSSQL instance.</td>
</tr>
<tr>
<td></td>
<td>Client Type</td>
<td>The type of the client connected.</td>
</tr>
<tr>
<td>SSIS</td>
<td>Job</td>
<td>The name of SSIS job can be empty.</td>
</tr>
<tr>
<td>SSIS file</td>
<td>Package file name</td>
<td>The name of the package file.</td>
</tr>
<tr>
<td>SSIS for MSSQL</td>
<td>Instance</td>
<td>The name of the MSSQL instance.</td>
</tr>
<tr>
<td></td>
<td>Client Type</td>
<td>The type of the client connected.</td>
</tr>
<tr>
<td>StoreParameters Components</td>
<td>Front</td>
<td>The parameters of the Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Farm</td>
<td>The farm host the Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Application name</td>
<td>The Citrix application name.</td>
</tr>
<tr>
<td>XenApp or Presentation Server</td>
<td>Type</td>
<td>The type of the Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Farm</td>
<td>The farm host the Citrix application.</td>
</tr>
<tr>
<td>Entry point type</td>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Icon path</td>
<td>The logic path of the Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Argument</td>
<td>The parameters of Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Path</td>
<td>The logic path of the Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Exec</td>
<td>The EXE name of Citrix application.</td>
</tr>
<tr>
<td></td>
<td>IP Address</td>
<td>The IP of host is member of Citrix farm.</td>
</tr>
<tr>
<td>Sybase</td>
<td>Instance</td>
<td>The name of Sybase instance.</td>
</tr>
<tr>
<td></td>
<td>Configuration File</td>
<td>The path to Tibco BW project.</td>
</tr>
<tr>
<td></td>
<td>Process Name</td>
<td>The name of Tibco BW main process.</td>
</tr>
<tr>
<td></td>
<td>Queue Port</td>
<td>The port of the queue used by the Tibco process.</td>
</tr>
<tr>
<td></td>
<td>Config IP</td>
<td>The IP of the source of the queue.</td>
</tr>
<tr>
<td>Tibco Conf File</td>
<td>Process name</td>
<td>The name of Tibco BW main process.</td>
</tr>
<tr>
<td></td>
<td>File</td>
<td>The path of the file is used by Tibco BW for integration.</td>
</tr>
<tr>
<td>Tibco Hawk</td>
<td>Configuration directory</td>
<td>The path to Tibco Hawk Configuration folder.</td>
</tr>
<tr>
<td>WMB Dependency</td>
<td>Command line</td>
<td>The command line of the WMB.</td>
</tr>
<tr>
<td>XenApp or Presentation Server Components</td>
<td>Farm</td>
<td>The farm host the Citrix application.</td>
</tr>
<tr>
<td></td>
<td>Icon path</td>
<td>The logic path of the Citrix application.</td>
</tr>
</tbody>
</table>

Provide entry points for mapping an application service

As an application service owner, you may receive a request for information about entry points in an email notification. Provide information about entry points to enable administrators to start discovery of an application service.

Role required: sm_app_owner or sm_admin

If information about application service entry points is missing, an administrator cannot start mapping the application service. In this case, the administrator sends a request for missing data from the application service form, which creates a service process task assigned to you. You receive an email notification with the link to the Questionnaire page where you must enter the missing data. The most important information is about the entry points, however, any additional data about business service components, their connections, or usage is of help for the administrator. When you finish entering data in the Questionnaire and submit it, the system closes the service process task.
1. Click the link in the notification email to access the Questionnaire page.
2. Alternatively, open to the Questionnaire page from the list of tasks assigned to you.
   a) Navigate to Service Mapping > Administration > My Tasks.
   b) Optional: Sort the list of service process tasks as required.
   c) Click the required task.
   d) Click the application service form link.
   e) Click Questionnaire in the left pane.
3. On the Planned Custom Entry Points tab, enter information about application service entry points.

   **Note:**
   If you cannot specify the exact attributes of the entry points, such as the URL or the IP address, add general information which can guide the administrator. For example, the type of the application, like Tomcat server.
4. To add information about any other components comprising this application service, click the Components tab, and enter the relevant data there.
5. When you finished entering the information, click Actions, and then select Submit Questionnaire.
   The data request task assigned to you closes. The application service state changes to In Progress.
6. If there is any other useful information concerning this application service, enter it in the **Notes** field under **Worknotes** and press Enter to post your comment.

**Mapping application services containing Citrix XenApp and Presentation Server**

Service Mapping can discover application services containing Citrix XenApp, Presentation Server, and Citrix Usage Collector.

### Discovered components

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citrix XenApp Server</td>
<td>Windows</td>
<td>6.x&gt;6.5, 7.x&lt;7.6</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Citrix XenApp or Presentation Server</td>
</tr>
<tr>
<td>Citrix Presentation Server</td>
<td>Windows</td>
<td>4.x, 5.x, 6.x&lt;6.5</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Citrix XenApp or Presentation Server</td>
</tr>
<tr>
<td>Citrix Usage Collector</td>
<td>Windows</td>
<td>7.x&gt;7.5, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Citrix Collector</td>
</tr>
<tr>
<td>Generic application based on the Icon Path</td>
<td>Windows</td>
<td>-</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Citrix Application Icon Path</td>
</tr>
</tbody>
</table>

Map application services containing Citrix XenApp and Presentation Server

Map an application service containing Citrix XenApps, Presentation Servers, and Citrix Usage Collectors and verify that Service Mapping discovered all Citrix components correctly.

- Provide a user with permissions to read and query the Citrix repository:
  - For versions 4.5 and later, a PowerShell script from the XenApp server.
  - For versions earlier than 4.5, a Microsoft Visual Basic Script (VBScript) from the Citrix Presentation Server

- On the Now Platform, configure a domain user for accessing the Windows OS as described in **Windows credentials**.

  **Note:** To use this credential later for XenAPP, set the **Credential alias** attribute to `cmdb_ci_appl_citrix_xenapp`. To use this credential later for Citrix, set the **Credential alias** attribute to `cmdb_ci_appl_citrix_collector`.

- (Optional) To discover using the WinRM protocol, see [Configure WinRM trusted hosts on MID Server](#).

- Configure a user with permissions to run PowerShell commands on the Citrix server.

- (For discovery of Citrix Presentation Server only) Find out the icon path for the entry point:
  1. In the Citrix Access Management Console, navigate to **Presentation Server** > **Applications**.
  2. Locate the Icon Path folder in the navigation tree.
  3. Right-click the Icon Path folder and click **Copy path**.

- (For discovery of Citrix XenApp Server only) Find out the icon path for the entry point by checking the Distinguish Name attribute for the Icon Path.
Role required: sm_admin

1. Create an application service:
   a) Navigate to Service Mapping > Application Services.
   b) Click New.
      The Discoverable by Service Mapping option is selected by default.
   c) In the Name field, enter a descriptive name for the application service.

2. Define the entry point for this application service:
   a) Select the Citrix XenApp tile.
   b) Define entry point attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>The URL of the Citrix XenApp or Presentation Server. The URL must contain a VIP IP or a physical host.</td>
</tr>
<tr>
<td>Port</td>
<td>The port that the Citrix XenApp or Presentation Server uses on the host.</td>
</tr>
<tr>
<td>Icon Path</td>
<td>The path to the logical application location as it appears in the Citrix Access Management Console. The path must start with the Applications folder. For example, /Application/Jane/Windows Media Player.</td>
</tr>
<tr>
<td>URL</td>
<td>The URL must contain a VIP IP or a physical host. For example, http://&lt;host&gt;/Citrix/StoreWeb.</td>
</tr>
</tbody>
</table>

c) Click Add.
d) Click Save.

3. Click Additional Info on the left and clear the Traffic based discovery check box.
4. Click Update.
5. Verify that the mapping result is satisfactory:
   a) Click View map and wait until the mapping is complete. It may take a few minutes.
   b) Review the application service map and make sure that it resembles in the following example.
If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery failed. The discovery log displays the message that starts with the following text: You do not have sufficient permissions.</td>
<td>The indication sections of the pattern fail, because the user does not have permissions for running PowerShell commands.</td>
<td>Make sure that you provided a user with permissions to run PowerShell commands on servers that host Citrix XenApp, Presentation Server, or Citrix Usage Collector.</td>
</tr>
</tbody>
</table>
Mapping application services containing Microsoft Exchange Server

Service Mapping can discover application services containing a Microsoft Exchange Server and the Exchange Server components.

Discovered components

If e-mail services in your organization deploy Microsoft Exchange Server, the discovered application service includes the following components:

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Active Directory Domain Services</td>
<td>Windows</td>
<td>13, 30, 31, 44, 47, 52, 56, 69</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Active Directory Service pattern</td>
</tr>
</tbody>
</table>
Typically, you do not want to discover Outlook clients connected to the Exchange Server to avoid noise. If this is the case, disable the traffic-based discovery for this application service.

For information on the discovery flow, refer to pattern-based discovery in Service Mapping. Failure to provide users with required permissions may cause Service Mapping to perform **traffic-based discovery**.

**Limitations**

Discovery and Service Mapping cannot discover the Exchange Server if its configuration does not support public folders, edge transports, and public messaging. Create a pattern or customize the existing pattern to discover the Exchange Server with such configuration.

Map application services containing Microsoft Exchange Servers

Map an application service containing an Microsoft Exchange Server and verify that Service Mapping discovered all Microsoft Exchange Server components correctly.

- On the Now Platform, configure a domain user for accessing the Windows OS as described in [Windows credentials](#). Use the Microsoft Exchange Server administrator's user name and password as Windows credentials. For example, user name: local\administrator and password: Are1300.

  **Note:** If you configure Windows credentials for a user different from the Microsoft Exchange Server administrator, Service Mapping uses traffic-based discovery that results in mapping all Microsoft Exchange clients.

- (For Exchange CAS 2010 and 2013, 2016) On the Now Platform, configure **applicative credentials**, setting the CI type to Microsoft Exchange Mailbox.

  **Warning:** Do not confuse the Exchange Mailbox CI type with the Microsoft Exchange Mailbox server CI type.

Service Mapping uses these applicative credentials to access the of/Servers/Servers.slab folder on the Microsoft Exchange Admin Center (EAC. Service Mapping also uses the applicative credentials to run the following PowerShell commands against the Microsoft Exchange servers:

- Get-ExchangeServer
- Get-ExchangeServer
- Get-MailboxServer
- Get-ClusteredMailboxServerStatus
- Get-ExchangeServer
- Get-StorageGroup
- Get-MailboxDatabase

- (For Microsoft Exchange CAS 2007, 2010) On the Now Platform, upload the rctrIx tool for running PowerShell commands as described in [Upload the rctrIx.exe file to MID Servers](#).
- (For Microsoft Exchange CAS 2007 and 2010) Provide an Exchange Mailbox OS user with the rights to run Exchange services on Windows and has permission to query Exchange repository:

  On the Now Platform, configure **Windows credentials** setting the **Credential alias** attribute to (cmdb_ci_exchange_mailbox). Configuring this attribute allows you later to use the credential for this CI in patterns.
• (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.

• Verify that you know the entry point necessary for discovering the Microsoft Exchange Server. Use the URL for the Outlook web access in the following format:
  http(s)://<hostname>:<port>/owa
  For example, http(s)://myhost:<443>/owa

Role required: sm_admin

1. Create an application service:
   a) Navigate to Service Mapping > Application Services.
   b) Click New.
      The Discoverable by Service Mapping option is selected by default.
   c) In the Name field, enter a descriptive name for the application service.

2. Define the entry point for this application service:
   a) Select the Web Application tile.
      The Discoverable by Service Mapping option is selected by default.
   b) In the URL field, enter the URL of the Outlook web access.
      For example, http(s)://myhost:<443>/owa.
   c) Click Add.
   d) Click Save.

3. Click Additional Info on the left and clear the Traffic based discovery check box.

4. Click Update.

5. Verify that the mapping result is satisfactory:
   a) Click View Map and wait until the mapping is complete. It may take a few minutes.
   b) Review the application service map and make sure that the service layout is as in the examples below.
If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery failed. The discovery log displays the message that starts</td>
<td>The indication sections of the pattern fail, because the user does</td>
<td>Make sure that you provided a user with permissions to run PowerShell</td>
</tr>
<tr>
<td>with the following text:</td>
<td>not have permissions for running PowerShell commands.</td>
<td>commands on servers that host Microsoft Exchange Server components, for</td>
</tr>
<tr>
<td>You must have the Exchange Common Commands PowerShell Snapin installed</td>
<td></td>
<td>example, CAS.</td>
</tr>
<tr>
<td>to use this script.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Discovery failed. The discovery log displays the error message that starts with the following text: (servername.destination.com) Connecting to remote server failed with the following error message: The WinRM client can not process the request. The authentication mechanism requested by the client is not supported by the server or unencrypted traffic is disabled in the service configuration. Verify the unencrypted traffic setting in the service configuration or specify one of the authentication mechanisms supported by the server.

The application service map is very large, because it includes connections to all Outlook clients. (In deployments using Microsoft Exchange Server 2007, 2010 or 2013), Service Mapping created connections using traffic-based discovery, which traces all open connections, instead of using PowerShell commands. Make sure that you provided a user with permissions to run PowerShell commands on servers that host Microsoft Exchange Server components, for example, CAS.

Mapping application services containing SAP applications
Service Mapping can discover application services containing SAP applications and their components.

Discovered components

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Business Application Programming (ABAP) SAP Central Services (ASCS)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP ASCS Application patterns</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>---------</td>
<td>----------</td>
<td>----------------</td>
<td>----------------------------------</td>
</tr>
</tbody>
</table>
| SAP Convergent Invoicing (CI), including:  
  - SAP Convergent Charging  
  - SAP Customer Relationship Management  
  - Contract Accounts Receivable and Payable | Windows, Linux, Hpux, Solaris, AIX | 7.x.x | WMI/WinRM/SSH | Horizontal and top-down | SAP CI Application pattern |
<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP NetWeaver Development Infrastructure (DI), including:</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP DI Application pattern</td>
</tr>
<tr>
<td>• Change Management Service (CMS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Design Time Repository (DRT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Component Build Service (CBS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• System Landscape Directory (SLD and Name Service)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Common user management of the development infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Integration of the transports with the Change Transport System (CTS) of Application Server ABAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Available tools for monitoring the development infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAP Evaluated Receipt Settlement (ERS)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP ERS Application pattern</td>
</tr>
<tr>
<td>SAP Java Connector (JC)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP JC Application pattern</td>
</tr>
<tr>
<td>SAP Central Services (SCS)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP SCS Application pattern</td>
</tr>
</tbody>
</table>
Limitations

Service Mapping can map only one instance of ASCS per deployment.

Map application services containing SAP applications
Map an application service containing an SAP applications and verify that Service Mapping discovered all SAP components correctly.

- Provide a user with permissions to run SOAP on RFC read table function.
- (If hosted on a Unix server) On the Now Platform, configure `applicative credentials` and `SSH credentials`. The SSH credentials are used to create connections from SAP Central Instance to other applications and devices. In addition, provide the user elevated rights to be able to run `Service Mapping commands requiring a privileged user`.
- (If hosted on a Windows server) On the Now Platform, configure `Windows credentials` to create connections from SAP Central Instance to other applications and devices.
- (Optional) For discovering using the WinRM protocol, see `Configure WinRM trusted hosts on MID Server`.
- (Optional) If your organization deploys SAP digital business services using web services and you want to map all connected business services, perform the following steps:
  - Configure applicative credentials for a user with permissions to run the following commands: CVERS, DBCONS, and RFCCONS.
  - Ensure that you have the web service URL. For information on how to create web services, see [https://blogs.sap.com/2012/10/24/how-to-create-web-services-abap-based/](https://blogs.sap.com/2012/10/24/how-to-create-web-services-abap-based/).

Role required: `sm_admin`

1. Create an application service:
   a) Navigate to `Service Mapping > Application Services`.
   b) Click `New`.
      - The `Discoverable by Service Mapping` option is selected by default.
   c) In the `Name` field, enter a descriptive name for the application service.

2. Define the entry point for this application service:
   a) Select the `SAP application` tile.
      - The `Discoverable by Service Mapping` option is selected by default.
   b) Define the entry point as follows:
      - A URL to any SAP module
      - A TCP connection to any SAP module
      - To discover environments with connected services, provide the URL to the web service in the following format:

```
http://<hostname>:<port>/sap/bc/srt/scs/sap/z_servicenow_read?sap-client=003
```

c) Click `Add`.

d) Click `Save`.

3. Click `Additional Info` on the left and clear the `Traffic based discovery` check box.
4. Click `Update`.
5. Verify that the mapping result is satisfactory:
a) Click **View Map** and wait until the mapping is complete. It may take a few minutes.
b) Review the application service map and make sure that the service layout is as in the following examples.

If the SAP environment integrates SAP components with Tibco services, you can expect this mapping result:
Troubleshoot application services containing SAP applications

Fix basic discovery issues based on the symptoms and discovery messages.

**Symptom**

On the Now Platform using Jakarta or any later release, the business service or application service map in Service Mapping displays the following message: Failed to discover <SAP component name>.
Cause

The CI failed to identify, and process detection retrieves the process attributes.

Solution

Perform the following steps:

1. Open the failed pattern step in Debug mode:
   a. On the business service or application service map, right-click the error message and select Show discovery log.
      The Discovery Log window opens.
   b. In the right pane, locate the failed identification section and the failed step within this section.
   c. Select this failed identification section and click Debug.
      The Debug Identification Section window opens.
   d. If necessary, fill in the required details and click Connect.
      For information about working in Debug mode, refer to Activate pattern Debug mode.

2. Click the set variables step under Steps.
3. Make sure that the regular expression includes SYS as shown in the following figure.
4. Click the `set install_dir` step under **Steps**.

5. Make sure that the regular expression includes `sap` as shown in the following graphic.
Symptom

Service Mapping fails to discover the web service your organization uses to deploy SAP digital business services.

Cause 1

The pattern fails at the EVAL function for identifying the web service.

Solution 1

1. Verify that the format of the entry point is the URL to the web service as described in the mapping procedure.
2. Verify that you configured the necessary applicative credentials as described in the mapping procedure.
3. Check that the discovery log contains the following message: GROOVE_CODE_FAILURE: No such property: com for class: Script1.
4. Open the failed pattern step in Debug mode:
   a. On the business service or application service map, right-click the error message and select Show discovery log.
      The Discovery Log window opens.
   b. In the right pane, locate the failed identification section and the failed step within this section.
   c. Select this failed identification section and click Debug.
      The Debug Identification Section window opens.
   d. If necessary, fill in the required details and click Connect.
      For information about working in Debug mode, refer to Activate pattern Debug mode.
5. Pattern Designer opens showing the selected identification section.
6. Select the failed step on the right.
7. Verify that the function running on EVAL is com.snc.sw.util.HttpInvokerUtil.post.
8. Try to map the application service again and check the result.

**Cause 2**

The pattern fails at the EVAL function for identifying the web service, because the web service was configured incorrectly.
Solution 2

1. Verify that the format of the entry point is the URL to the web service as described in the mapping procedure.

2. Verify that you configured the necessary applicative credentials as described in the mapping procedure.

3. Check that the discovery log contains the following message: "Debug run railed: Test step failed. GROOVE_CODE_FAILURE: All command implementations (HTTP) failed on host 10.x.x.x. Host must be down or unreachable. ()"

4. Recreate the web service as described in https://blogs.sap.com/2012/10/24/how-to-create-web-services-abap-based/.

5. Try to map the application service again and check the result.

Cause 3

The pattern fails at the EVAL function for identifying the web service, because the URL configured in the entry point is not correct or because the MID Server cannot access this URL.

Solution 3

1. Verify that the format of the entry point is the URL to the web service as described in the mapping procedure.

2. Verify that you configured the necessary applicative credentials as described in the mapping procedure.

3. Check that the discovery log contains the following message: "Debug run railed: Test step failed. GROOVE_CODE_FAILURE: All command implementations (HTTP) failed on host 10.x.x.x. Host must be down or unreachable. ()"

4. Verify that you used the correct URL for the service entry point:
   a. In the TCode SOAMANAGER, navigate to Simplified Web Service Configuration.
   b. On the Configuration tab of the Simplified Service Configuration window, search for the web service that is part of the application service.
   c. Click the check box under User Name/Password (Basic) for this web service.
   d. Click Show Details.
   e. On the Details of the Service Definition window, verify the actual URL of the web service under WSDL URLs.
   f. If necessary, fix the URL you configure for the entry point and try to map the application service again.

Cause 4

MID Server cannot access this URL configured in the entry point of the application service.
Solution 4

1. Verify that the format of the entry point is the URL to the web service as described in the mapping procedure.
2. Verify that you configured the necessary applicative credentials as described in the mapping procedure.
3. Check that the discovery log contains the following message: "Debug run failed: Test step failed. GROOVE_CODE_FAILURE: All command implementations (HTTP) failed on host 10.x.x.x. Host must be down or unreachable. ()"
4. Verify that you used the MID Server that can access the URL you configured in the service entry point:
   a. Log into the MID Server host remotely.
   b. Open a browser.
   c. Paste the URL into the browser address bar and press Enter.
   d. If the page does not open, it means that there is a network issue, for example, there is a firewall that prevents the MID Server from accessing this URL.
5. Resolve the network issue and verify that the MID Server can access the URL as described in step 4.

Cause 5

MID Server cannot access this URL configured in the entry point of the application service.

Solution 5

1. Verify that the format of the entry point is the URL to the web service as described in the mapping procedure.
2. Verify that you configured the necessary applicative credentials as described in the mapping procedure.
3. Check that the discovery log contains the following message: "Debug run railed: Test step failed. GROOVE_CODE_FAILURE: All command implementations (HTTP) failed on host 10.x.x.x. Host must be down or unreachable. ()"
4. Verify that the applicative credentials you configured on the Now Platform for the web service are valid:
   a. Log into the MID Server host remotely.
   b. Open a browser.
   c. Paste the URL into the browser address bar and press Enter.
   d. The browser page displays the credentials pop-up window.
   e. Enter the credentials and check that the page opens.
      The page does not open, it means that the configured applicative credentials are incorrect.
5. Find out the right credentials.
6. Verify the new credentials as described in step 4.

7. If the page opens after you use the new credentials, configure these credentials as applicative credentials on the Now Platform.

**Cause 6**

MID Server cannot access this URL configured in the entry point of the application service.

**Solution 6**

1. Verify that the format of the entry point is the URL to the web service as described in the mapping procedure.

2. Check that the discovery log contains the following message: ‘Debug run railed: Test step failed. GROOVE_CODE_FAILURE: Missing applicative credentials for CI type <name of the CI type>”.

3. Verify that you defined applicative credentials for the relevant CI according as described in Map application services containing SAP applications.

**Mapping application services containing TIBCO BusinessWorks and EMS**

Service Mapping can discover application services containing TIBCO BusinessWorks, Enterprise Message Service (EMS), and their components.

If services in your organization deploy TIBCO applications, the discovered application service includes the following components:

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIBCO Enterprise Message Service (EMS)</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Enterprise Message Service</td>
</tr>
<tr>
<td>TIBCO Enterprise Message Service (EMS) Queue</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Enterprise Message Service</td>
</tr>
</tbody>
</table>

Map application services containing TIBCO BusinessWorks and EMS

Map an application service containing TIBCO BusinessWorks and Enterprise Message Service (EMS) and verify that Service Mapping discovered all TIBCO components correctly.

- (If hosted on Unix) Configure **applicative credentials** for the TIBCO EMS.
- (If hosted on a Windows server) On the Now Platform, configure **Windows credentials**.
- (If hosted on a Unix server) On the Now Platform, configure **SSH credentials**.
• (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.

Role required: sm_admin

1. Create an application service:
   a) Navigate to Service Mapping > Application Services.
   b) Click New.
      The Discoverable by Service Mapping option is selected by default.
   c) In the Name field, enter a descriptive name for the application service.

2. Define the entry point for this application service:
   a) Select the Other application tile.
      The Discoverable by Service Mapping option is selected by default.
   b) From the Select Entry Point Type list, select HTTP(s) Endpoint, TCP Endpoint, Tibco BW Endpoint.
   c) Define the entry point as follows:
      • The URL to the process that correlates to the TIBCO BusinessWorks Project, for example, http://10.xx.xx.xx:6710/GetMemberActivitiy/Processes/ServiceGetMemberActivitiy.service.
      • (Optional) The hostname on which the BusinessWorks Project runs as a service.
   d) Click Add.
   e) Click Save.

3. Click Additional Info on the left and clear the Traffic based discovery check box.

4. Click Update.

5. Verify that the mapping result is satisfactory:
   a) Click View Map and wait until the mapping is complete. It may take a few minutes.
   b) Review the application service map and make sure that the service layout is as in the following examples.
In addition, verify that the BusinessWorks CIs are inclusions each of which contain a BusinessWorks process. Also, verify that the EMS CIs are inclusions containing EMS Queues.
If the mapping result is not as you expected, perform basic troubleshooting.
### Symptom

<table>
<thead>
<tr>
<th>(Only for deployments hosting BusinessWorks on a Unix server) The application service map shows the BusinessWorks CI without an inclusion of the BusinessWorks process. The discovery log shows the error message stating that the user does not have sufficient credentials.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Only for deployments hosting BusinessWorks on a Unix server) The application service map shows the BusinessWorks CI without an inclusion of the BusinessWorks process. The discovery log shows the error message stating that the user does not have sufficient credentials.</td>
</tr>
<tr>
<td>Service Mapping failed to run the <code>TibcoFilesParser.ksh</code> command.</td>
</tr>
<tr>
<td>Make sure that you provided a user with elevated rights as described in <code>Service Mapping commands not requiring a privileged user</code>.</td>
</tr>
<tr>
<td>Service Mapping failed to discover the outgoing connections of EMS to the EMS process.</td>
</tr>
<tr>
<td>Service Mapping failed to run the <code>tibemsadmin</code> command.</td>
</tr>
<tr>
<td>Make sure that you provided a user with elevated rights as described in <code>Service Mapping commands not requiring a privileged user</code>.</td>
</tr>
</tbody>
</table>

### Mapping application services containing Oracle GlassFish Servers

Service Mapping can discover application services containing Oracle GlassFish Servers.

### Discovered components

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle GlassFish Server</td>
<td>Windows/Unix</td>
<td>3.1, 4.0, 4.1</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>GlassFish Server</td>
</tr>
<tr>
<td>Oracle GlassFish WAR file</td>
<td>Windows/Unix</td>
<td>3.1, 4.0, 4.1</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>GlassFish WAR</td>
</tr>
</tbody>
</table>

Map application services containing Oracle GlassFish Servers.

Map an application service containing an Oracle GlassFish Server correctly.

- (If hosted on a Windows server) On the Now Platform, configure Windows credentials.
- (If hosted on a Linux or Unix server) (Optional) on the Now Platform, configure SSH credentials.
- (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.

Role required: sm_admin

1. Create an application service:
   a) Navigate to Service Mapping > Application Services.
   b) Click New.
      The Discoverable by Service Mapping option is selected by default.
   c) In the Name field, enter a descriptive name for the application service.

2. Define the entry point for this application service:
   a) Select the Other Application file.
      The Discoverable by Service Mapping option is selected by default.
b) Define the entry point by filling in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Select Entry Point Type | Entry point type:  
  - HTTP(S) Endpoint  
  - TCP Endpoint  
  - Tibco BW Endpoint |
| URL                    | The URL of the process that correlates with the GlassFish Server, for example, http://10.196.xx.xx:28080/hello/.  
  This field appears when HTTP(S) Endpoint is selected in Select Entry Point Type. |
| Host                   | The server that hosts the GlassFish Server.  
  This field appears when Tibco BW Endpoint or TCP Endpoint is selected in Select Entry Point Type. |
| Host Name              | The host name of the server that hosts the GlassFish Server.  
  This field appears when HTTP(S) Endpoint is selected in Select Entry Point Type. |
| Port                   | The port, which the process correlating with the GlassFish Server uses.  
  This field appears when Tibco BW Endpoint or TCP Endpoint is selected in Select Entry Point Type. |
| Process Name           | The name of the process that correlates with the GlassFish Server.  
  This field appears when Tibco BW Endpoint is selected in Select Entry Point Type. |
| Comments               | Free text comment to provide information on the endpoint.  
  This field appears when HTTP(S) Endpoint or TCP Endpoint is selected in Select Entry Point Type. |

c) Click Add.

d) Click Save.

3. Click Additional Info on the left and clear the Traffic based discovery check box.
4. Click Update.
5. Verify that the mapping result is satisfactory:
   a) Click View Map and wait until the mapping is complete. It may take a few minutes.
   b) Review the application service map and make sure that the service layout is as in the following example.
In addition, verify that the GlassFish Server CIs are inclusions, each of which contains a WAR file.

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
</table>
| Service Mapping fails to discover the GlassFish Server CI. The discovery log displays the following message: Process is not running or port is not correct. Failed to identify CI due to the host 10.196.xx.xx lacking an application running on port 28080. Make sure that the application is running and run quick discovery on the host. | There is no process running on the port that is part of the entry point. | 1. Make sure that the entry point is correct.  
2. If the entry point is correct, check that there is a process running on the port by running a quick discovery on the host. |

Mapping application services containing Oracle Tuxedo platform

Service Mapping can discover application services containing Oracle Tuxedo platform and its components.
Map application services containing Oracle Tuxedo platform

Map an application service containing an Oracle Tuxedo platform and verify that Service Mapping discovered all Tuxedo components correctly.

- On the Now Platform, configure *SSH credentials*.
- (Optional) For discovering using the WinRM protocol, see *Configure WinRM trusted hosts on MID Server*.
- (If hosted on Unix) Provide a user with the elevated rights for running commands described in *Service Mapping commands requiring a privileged user*.

Role required: sm_admin

1. Create an application service:
   a) Navigate to Service Mapping > Application Services.
   b) Click New.
      The Discoverable by Service Mapping option is selected by default.
   c) In the Name field, enter a descriptive name for the application service.

2. Define the entry point for this application service:
   a) Select the Other Application tile.
   b) Select TCP Endpoint from the list.
   c) Define the entry point as follows:
      - A host of the server that hosts the Tuxedo process.
      - A port on this server that the Tuxedo process opens.
   d) Click Add.
   e) Click Save.

3. Click Additional Info on the left and clear the Traffic based discovery check box.
4. Click Update.
5. Verify that the mapping result is satisfactory:
   a) Click View Map and wait until the mapping is complete. It may take a few minutes.
   b) Review the application service map.

If the mapping result is not as you expected, perform basic troubleshooting.
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Mapping fails to discover the Tuxedo CI. The discovery log</td>
<td>The provided user does not have elevated permissions for running</td>
<td>Make sure that you provided a user with permissions to run commands requiring elevated rights.</td>
</tr>
<tr>
<td>displays the message that states that the credentials are insufficient</td>
<td>this command.</td>
<td></td>
</tr>
<tr>
<td>for running the command including “permissions.tmadmin -v”.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Mapping fails to discover the Tuxedo CI. The discovery log</td>
<td>The provided user does not have the permissions to run the tux.env</td>
<td>Make sure that you provided a user with permissions to run the tux.env script.</td>
</tr>
<tr>
<td>displays the message that states that the user cannot run the tux.env</td>
<td>script.</td>
<td></td>
</tr>
<tr>
<td>script.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Fix application service errors in bulk

Service Mapping classifies errors by their root cause, for example, missing credentials or task timeout. For a fast and efficient process, fix errors belonging to the same category in bulk.

**Role required:** sm_admin or admin

You can view errors by category, for example, Configuration, Network, and Credentials. Each category tile also displays the error counter.

As part of the Service Mapping workflow, after you map application services in bulk, you can fix multiple application service errors in bulk (as a group).
After performing initial mapping, Service Mapping rediscovers configuration items (CIs) regularly, constantly updating the list of errors. Discovery schedules define how often Service Mapping runs the discovery process for CIs. Service Mapping discovers only CIs belonging to application services with Operational status.

There may be discovery errors not assigned to any category:

- After upgrading Service Mapping from Jakarta release or earlier.
If there are configuration items (CIs) that have errors associated with them, and that were first removed from application services and then readded to application services.

1. Navigate to **Service Mapping > Home** and then click **Fix Your Services** in the **Fix** box.

   The number in the **Fix** box indicates the number of affected service maps. Click **Update** to update the display.

   **Note:** To fix the errors for a particular service instead, click **Manage Errors by Service**.

2. If there are uncategorized errors, you can categorize them by application service:
   a) Click **View Uncategorized Errors by Service**.
   b) Click the relevant service name.
   c) On the application service map, click **Run Discovery**.

   Service Mapping discovers all CIs discovered with errors and belonging to this application service.

3. If there are uncategorized errors, categorize them by clicking **Rediscover Relevant CIs**, and clicking **Continue** in the confirmation message.

   Service Mapping discovers all CIs discovered with errors and belonging to application services with Operational status.

4. Click an error category box.

   In the example below, configuration errors are selected.

   - The Errors by Category page organizes errors by high, medium, and low impact on the services.
   - For each impact category, errors are grouped into boxes by error code (possibly the root cause).
   - If error tasks have been assigned, the task numbers appear in the box. Click a task to view details. Point to a task to view comments that the assigner might have added. A check mark indicates that a task is completed.
5. To view the list of errors that share an error code, click the box. In the example, All patterns failed is selected.

You can perform one of the Recommended Actions on items that you select or on all items in the list.
6. Click an action in the Action on All section, or select errors and click an action in the Action on Selected section.

**Retry Discovery**
Performs the full Discovery process. You typically perform Discovery after following the fix instructions that appear when you click View instructions.

**View affected services**
Displays a popup list of all services that are affected by the selected error. In the list, click a service to view service details.

**Mark as assigned**
Apply the Mark as assigned action when you plan to perform an out-of-band action to notify a person to fix the error (typically a person who does not use the ServiceNow instance). You can enter a Comment in the Mark as Assigned popup. Click Update to set the Error Status field in the error record as Assigned and associate the Comment with the error. To view the comment, point to the Assigned link.

Note: The Create ServiceNow task action also sets the Error Status field in the error record as Assigned.

**Create a ServiceNow task**
Opens the ServiceNow Error task form in a pop-up window so you can create and assign a task for the selected errors. Sets the Error Status field in the error record as Assigned. For information on creating error tasks, see Task table and Create a task from an incident, problem, or change request.

Note: The Mark as assigned action also sets the Error Status field in the error record as Assigned.
**Retry Discovery**

Performs the full Discovery process. You typically perform Discovery after following the fix instructions that appear when you click **View instructions**.

**View instructions**

Displays a troubleshooting procedure in a pop-up window. Follow the steps to try to resolve the error.

**Ignore Error**

Right-click an error and then select **Ignore Error** to set the **Ignored** value to **true**.

The system performs the action on the items, displays **In progress**, and then displays the result (success or failure) of the action. If the error is resolved, the **Error Status** field in the error record is set to **Resolved**.
Note: If an action (for example, Discovery) is in progress, but taking too long, you can click **Stop** to stop the action.

7. Click **Done** to close the status message.
8. If you performed an action on a selection and the error was resolved, you can click **Resolve All** to apply the same action to all items in the list.

**Fix errors in individual application service maps**

You can address discovery issues in each application service map individually.

Role required: sm_admin

You can fix errors in individual application services at any time. When you perform initial mapping of application services in your organization, you typically fix errors in individual application services after resolving errors in multiple services, in bulk. You may also need to fix errors in an approved and completed application service at a later stage.
1. Alternatively, if Service Mapping is deployed, navigate to **Service Mapping > Services > Application Services**.
2. Click **View Map** next to the relevant application service.
3. Ensure that the map opens in **Edit mode**.
CIs discovered with errors appear with the warning icon (⚠️) on the map. Discovery messages on the **Discovery Messages** tab give short error descriptions.

4. Optional: Group errors by types by clicking **Group Errors**.
   The discovery error messages on the **Discovery Messages** tab appear grouped by predefined types, error codes.

5. To review and fix an error from the list, click **Handle Error** next to the error description.
   Or

6. To review the first error in the largest error group, click **Handle Next Error** at the left top corner of the map.

7. To learn how to resolve the error, click **Open instructions** under **Action on Selected** in the right pane.

   **Note:** Instructions are available only for the most common errors.

   The Instructions window opens.

8. Click the relevant semi-automated resolution option to resolve the error under **Action on Selected** in the right pane.
Error resolution options vary, because they depend on the specific discovery error associated with the selected CI.

<table>
<thead>
<tr>
<th>Use this option</th>
<th>To do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add Management IP</td>
<td>Define the management IP for the server hosting the application, which Service Mapping failed to discover.</td>
</tr>
<tr>
<td>Create new WMI credentials</td>
<td>Open the Windows Credentials form and configure missing Windows credentials for the selected CI. See Windows credentials.</td>
</tr>
<tr>
<td>Create new SSH credentials</td>
<td>Open the SSH Credentials form and configure missing SSH credentials for the selected CI. See SSH credentials.</td>
</tr>
<tr>
<td>Create new SSH private key</td>
<td>Open the SSH private keys form and configure missing credentials for the selected CI. See SSH private key credential type.</td>
</tr>
<tr>
<td>Create new SNMP credentials</td>
<td>Open the SNMP credentials form and configure missing SNMP credentials for the selected CI. See SNMP community credentials.</td>
</tr>
<tr>
<td>Show Discovery Log</td>
<td>See the log to understand at what stage an error occurred and what caused it.</td>
</tr>
<tr>
<td>Skip and resume</td>
<td>Manually add the CI whose discovery failed and configure attributes of the entry point from which Service Mapping continues discovery. For more information, see Skip errors to continue discovering an application service.</td>
</tr>
<tr>
<td>Retry discovery</td>
<td>Run the discovery process for the selected CI after trying to resolve a discovery error associated with it.</td>
</tr>
<tr>
<td>View affected services</td>
<td>Display a list of all services affected by the selected error. In the list, click a service to view service details.</td>
</tr>
</tbody>
</table>

9. If there are no instructions or semi-automated resolution options available, fix errors based on symptoms and discovery messages.

10. To ignore an irrelevant error, right-click its error message on the Discovery Messages tab and select Toggle Ignore State. The ignored error disappears from the map and from the list of discovery error messages.

Note: If necessary, you can still display ignored errors by switching on the Include Ignored Errors option under More Options menu. For more info, see Modify view for an application service map.

11. Click Handle Next Error to review the next error in the largest error group.

12. When finished reviewing and handling errors, click anywhere in the map to close the Resolve Error pane.

If you resolved most errors, send the application service for review.

Fix errors in individual application services using discovery messages

Service Mapping does not offer semi-automated resolution options for errors that require advanced resolution. Fix such errors using symptoms and discovery messages.
Try fixing errors using semi-automated resolution options as described in the *Fix errors in individual application service maps*.

Role required: sm_admin

You can fix errors in individual application services at any time.

1. Navigate to **Service Mapping > Services > Application Services**.
2. Click **View Map** next to the relevant application service.
3. Ensure that the map opens in Edit mode.

CIs discovered with errors appear with the warning icon (⚠️) on the map. Discovery messages on the **Discovery Messages** tab give short error descriptions.

4. Fix errors based on the symptoms and discovery messages.
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>The application service map displays the warning icon (⚠️) on top or instead of the configuration item.</td>
<td>KB0621669: Resolving failure to execute commands using sudo during Discovery in Service Mapping</td>
</tr>
</tbody>
</table>
| The following discovery message displays for the configuration item: Failed to execute command using sudo on host `<host IP address>`.
<p>| | |
|                                                                                                 |                                                                                             |
| The following error message displays for the configuration item: Access is denied.                 | KB0564283: Resolving an issue of denied access to a Windows Server                           |
|                                                                                                 |                                                                                             |
| The application service map displays the warning icon (⚠️) on top or instead of the Windows Server. | KB0564296: Resolving failure to run commands on Windows Servers during discovery in Service Mapping |
| The following discovery message displays for the Windows Server: Failed to execute WMI command on host.  |                                                                                             |
|                                                                                                 |                                                                                             |
| The application service map displays the warning icon (⚠️) on top or instead of the Windows Server. | KB0564282: Resolving issue of RPC Server unavailable during Windows Server discovery in Service Mapping |
| The following discovery message displays for the Windows Server: RPC Server unavailable.           |                                                                                             |
|                                                                                                 |                                                                                             |
| The application service map displays the warning icon instead of the load balancer CI.             | KB0610414: Resolving failure to discover operating system for a load balancer                  |
| The following discovery message is displayed: Service Mapping triggered the horizontal discovery to find the host <code>x.x.x.x</code>, because this host was not in the CMDB. The horizontal discovery failed. See discovery status for more info. |
|                                                                                                 |                                                                                             |
| Instead of the map, the following discovery message displays: Cannot display the map. Topology too large. | KB0596671: Resolving a large topology issue in application service maps                         |
|                                                                                                 |                                                                                             |
| A Virtual IP (VIP) for a load balancer service CI is sometimes updated with the IP addresses of another load balancer. | KB0610412: Resolving load balancers merging into one CI in Service Mapping                     |
|                                                                                                 |                                                                                             |
| The map displays incorrect virtual IP names for load balancers.                                  | KB0610413: Resolving incorrect virtual IP names of load balancers in Service Mapping           |
|                                                                                                 |                                                                                             |
| The application service map consists of different CIs every time you run the mapping process on the same entry points to discover the same application service. | KB0595227: Troubleshooting inconsistent mapping results                                         |</p>
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>There is no load balancer CI on the application service map. There are connections going directly from the entry point of the application service to the next tier CIs. The rest of the CIs also disappear from the map.</td>
<td>KB0621529: Resolving the issue of a load balancer disappearing from the business service map</td>
</tr>
<tr>
<td>- The application service map displays the warning icon (⚠️) on top or instead of the configuration item.</td>
<td>KB0621670: Resolving the issue of SSH command time-out during discovery in Service Mapping</td>
</tr>
<tr>
<td>- The following discovery message displays for the configuration item: SSH command timed out on host.</td>
<td>KB0621531: Resolving disconnected cluster CIs on the application service map</td>
</tr>
<tr>
<td>A cluster CI appears detached from all CIs in map tiers above or below it.</td>
<td>KB0621576: Resolving failure to discover VIP for a load balancer</td>
</tr>
<tr>
<td>- The map displays either the load balancer configuration item (CI) with a warning icon or just the warning icon</td>
<td>KB0621616: Business service map displays a wrong load balancer in Service Mapping</td>
</tr>
<tr>
<td>- The following discovery message appears for the CI that is expected to be the load balancer in the application service: Failed to recognize application. See the discovery log for more details.</td>
<td>KB0621670: Resolving the issue of SSH command time-out during discovery in Service Mapping</td>
</tr>
<tr>
<td>The application service map shows a different load balancer from the one you expected to see in this business service.</td>
<td>KB0621673: Resolving a failure to communicate with the WMI Collector during discovery in Service Mapping</td>
</tr>
<tr>
<td>- The application service map displays the warning icon (⚠️) on top or instead of the configuration item.</td>
<td>KB0621673: Resolving a failure to communicate with the WMI Collector during discovery in Service Mapping</td>
</tr>
<tr>
<td>- The following discovery message displays for the configuration item: SSH command timed out on host.</td>
<td>KB0621673: Resolving a failure to communicate with the WMI Collector during discovery in Service Mapping</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>The application service map displays the warning icon (⚠️) on top or</td>
<td><strong>KB0622808: Resolving a failure to find a MID Server for an IP address</strong></td>
</tr>
<tr>
<td>instead of the configuration item.</td>
<td>Perform the procedure described in <a href="#">Upload the rctrlx.exe file to MID Servers</a>.</td>
</tr>
<tr>
<td>The following discovery message displays for the configuration item:</td>
<td></td>
</tr>
<tr>
<td><em>No active MID Server found for IP X.X.X.X.</em></td>
<td></td>
</tr>
<tr>
<td>'64\rctrlx' is not recognized as an internal or external command,</td>
<td></td>
</tr>
<tr>
<td>operable program or batch file.</td>
<td></td>
</tr>
<tr>
<td>A CI or a segment of the map is grayed out after Service Mapping runs</td>
<td>1. (Optional for grayed out segments)</td>
</tr>
<tr>
<td>the top-down discovery on a previously mapped application service.</td>
<td>Identify the faulty CI that causes the segment to be grayed out.</td>
</tr>
<tr>
<td>In this business service Service Mapping failed to rediscover the CI</td>
<td>2. Use the discovery message associated with this faulty CI to resolve the problem.</td>
</tr>
<tr>
<td>or connections leading to this CI.</td>
<td></td>
</tr>
</tbody>
</table>

5. Click **Run Discovery**.

**Skip errors to continue discovering an application service**

If you know what configuration items (CIs) and connections make up your application service, you can enable Service Mapping to continue discovery of the application service even if there are some errors. You can skip errors to troubleshoot later so you can complete mapping most of the application service, even if some CIs are missing.

Familiarize yourself with the structure of the application service to know which CIs make up the next tier after the problematic CI. If necessary, consult the application service owner to collect necessary information.

Role required: sm_admin or admin

Normally, discovery of an application service stops when Service Mapping fails to identify a CI. In this case, the map shows a warning icon instead of the undiscovered CI. There are no CIs or connections going out of the undiscovered CI.
You can overcome the obstacle of the undiscovered CI and restart discovery from the CIs belonging to the next tier after the problematic CI. To do so, add a CI manually if you know what the undiscovered CI is and what the incoming connection of the next tier CI is. Service Mapping uses incoming connections to access a CI for discovery. These connections are referred to as endpoints.

1. Navigate to Service Mapping > Services > Application Services.
2. Click View map next to the application service that you want to view.
3. Ensure that the map opens in Edit mode.
4. Ensure that the map is not in the Host View.
5. Click the arrow next to the undiscovered CI and select **Skip and Resume**.
6. Define attributes of the CI you want to manually place instead of the undiscovered CI.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Select the CI type (CI class) for the CI you are adding. Every CI belongs to a CI type which contains a set of attributes configured for this kind of CI, for example, <code>cmdb_ci_appl</code> for applications.</td>
</tr>
<tr>
<td>CI Name</td>
<td>Select the CI from the list of CIs of the selected CI type.</td>
</tr>
</tbody>
</table>

**Note:**
If the CI type does not appear in the list, Service Mapping does not allow adding CIs of this type manually. For example, you cannot add an application cluster or an endpoint manually.
7. Configure attributes of the entry point from which Service Mapping continues discovery by selecting the entry point type and entering relevant fields. If the required entry point type does not appear in the Select Entry Point Type list, create it as described in Create entry point types for Service Mapping.

8. Click Submit.

The map shows the manually added CI instead of the undiscovered CI. Service Mapping starts discovery for CIs of the next tiers. When the discovery process finishes, the map shows all discovered CIs below the manually added CI.

To discover the problematic CI automatically:

1. Perform necessary troubleshooting to enable Service Mapping to discover the problematic CI. For example, you may need to customize the relevant pattern, provide missing credentials or fine-tune traffic-based discovery.

2. Navigate to the relevant application service map.

3. Click Run discovery.

4. After the discovery process finishes, verify that Service Mapping discovered the CI. If Service Mapping discovered the CI, the map shows the actual CI without the Warning icon (⚠).

**Resolve pattern-related mapping errors**

You can troubleshoot mapping errors caused by patterns.
Verify that the mapping error is caused by an inaccurate pattern by checking the discovery message. If the message says “Failed to recognize application”, the error is pattern-related.

You need to be familiar with the Pattern Designer module of Service Mapping.

Role required: admin and sm_admin

If there are configuration items (CIs) that Service Mapping could not map correctly, they appear on your application service map with warning icons. Typically, mapping errors happen when Service Mapping fails to connect to a CI or fails to recognize it due to an inaccurate pattern.

You can identify problematic steps in your pattern and fix them without reviewing all steps and operations contained in the pattern. It allows you to troubleshoot pattern-related errors quickly and effectively.

The following video provides an alternative way of troubleshooting pattern-related errors:

1. Navigate to **Service Mapping > Services > Application Services**.
2. Click **View map** next to the application service that you want to view.
3. Ensure that the map opens in Edit mode displaying discovery messages and errors.

4. Right-click the CI with the warning icon (⚠️) and select **Show discovery log**. The Discovery Log window opens showing mapping patterns and their details.
5. Expand the failed pattern which appears above the result marked red.

6. Expand the failed section and click the failed step.
7. Click **Debug**.
The Pattern Designer window opens showing the pattern with its steps in the left pane.

8. Click the first step and wait for Pattern Designer to run it. Repeat this action for other steps following the order until you reach the faulty step that causes the error.

In the example below the port number is 8081 instead of 8080 which causes the problem.

9. Click **OK**.

10. Fix the pattern attribute that causes the problem.

11. Click **Test**.

12. Verify that Pattern Designer returns a success message.

13. Click **Save**.

14. Click **Activate**.

15. Return to the application service map which contained the mapping error.

16. Right-click the CI for which the pattern was fixed and select **Resume discovery**.

17. Verify that the CI is discovered and mapped correctly.

---

**View status for application services discovered with Service Mapping**

The dashboard provides an up-to-date overview of the mapping status of all application services discovered with Service Mapping, progress on error tasks (tasks that are associated with mapping errors), and counts of error tasks and other tasks associated with services.
Role required: sm_app_owner or sm_admin

Navigate to Service Mapping > Home and click Map Status.

The scorecards count error tasks plus all other tasks. The scorecards and charts are standard ServiceNow widgets that you can modify to suit your needs. Click a data point to view the supporting data.
Review and approval of application service maps

After the Service Mapping administrator maps IT services and fixes errors in them, the administrator and the owner collaborate to review and approve the IT service maps.

You must have Service Mapping enabled to map or review application services.

Users who perform review and approval have the roles of a Service Mapping administrator and an application service owner. The Service Mapping administrator is responsible for mapping, fixing, and maintaining Application services. The application service owner checks that the application service maps are complete and all major components comprising it are correctly represented. If necessary, the owner leaves comments, referred to as reject messages, on application service maps for the Service Mapping administrator to implement.

The review and approval process is performed in the following stages:

1. After the initial mapping of application services, the administrator fixes errors and sends individual application services for review. The system creates a service process task assigned to the application service owner and sends an email notification about it.

2. The application service owner checks that the application service maps are complete and all major components comprising it are correctly represented. If necessary, the owner leaves comments, referred to as reject messages, on application service maps for the Service Mapping administrator to implement. The service process task assigned to the owner closes. The system sends an email notification to the administrator that the owner posted comments.

3. The administrator receives the email notification, views owner requests and implements them.

4. The administrator resends the application service maps for review. The system again creates a service process task assigned to the application service owner and sends an email notification about it.

5. If the revised application service maps are satisfactory, the application service owner approves them. If not, the owner requests further fixes, which the administrator must address.

To see documentation for a review phase, click the relevant box in the diagram.
1. **Send the application service for review**
2. **Review the business service and request fixes**
3. **Review and implement owner’s requests**
4. **Resend the application services for review**
5. **Review and approve the application service**

The process of application service review may take some time as it requires making changes and repeatedly running the mapping process on the application service. Typically, it takes several iterations to arrive at the desired result.

Once the owner approves an application service, the application service status changes to Approved and it appears in the list of completed application services on the **Home** page.

**Send application service maps for review**

After you map an application service, send it to the application service owner for review to make sure that the map is accurate.

Perform initial error fixing as described in [Fix application service errors in bulk](#) and [Fix errors in individual application service maps](#).
Role required: sm_admin

While you can fix errors in bulk, you always send application services for review individually, one by one.

Sending application service maps for review is part of the review and approval process. Typically, you send each application service map for review twice: The first time for the initial owner review and the second time after you implemented owner feedback.

After the initial mapping of application services, you fix errors and send individual application services for review. The system creates a service process task assigned to the application service owner and sends an email notification about it.

To see documentation for another review phase, click the relevant box in the diagram.

1. **Review the application service and request fixes**
2. **Review and implement owner’s requests**
3. **Resend the application services for review**
4. **Review and approve the application service**

1. To send an application service for review from the application service form:
   a) Navigate to Service Mapping > Services > Application Services.
   b) Sort the list of planned application services by status and scroll to see services in **In progress** status, or use the filter to narrow the list.
   c) Click the required application service.
   d) Click **Actions > Send For Review**.

2. Alternatively, send an application service for review from its map:
   a) Navigate to Service Mapping > Home. The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.
   b) Click the **Approve** tile.
   c) Click **Send For Review**.
   d) Click the required application service. The map for this application service opens.
   e) Click **Send For Review** in the upper right corner of the window.

When you receive an email notification that the owner sent comments for this application service, **review and implement the owner's requests**.

**Review application service maps**

As the application service owner for the application service map, you receive an email notification that the application service map is assigned to you for review. Review mapping results for correctness and either provide your feedback or approve the application service map.

Role required: sm_app_owner

Ideally, only the application service owner reviews the planned application service, however, users with the sm_app_owner or sm_admin role can also perform this task.

Reviewing application service maps is part of the **review and approval process**. The process of application service review may take some time as it requires making changes and repeatedly running the mapping process on the application service. Typically, it takes several iterations to arrive at the desired result. Once you are satisfied with the discovery result, you approve the application service.

After you request fixes or approve the application service, the service process task assigned to you closes.

To see documentation for another review phase, click the relevant box in the diagram.
1. **Send the application service for review**
2. **Review and implement owner's requests**
3. **Resend the application services for review**

1. Click the link to the application service map in the email notification. The map of the application service opens in the ServiceNow instance.
2. Alternatively, open to the application service to review from the list of tasks assigned to you.
   a) Navigate to **Service Mapping > Administration > My Tasks**.
   b) Optional: Sort the list of service process tasks as required.
   c) Click the required task.
   d) Click the application service form link.
   e) Click **View map**.
3. On the application service map, check that all essential CIs comprising the application service are discovered and mapped correctly.
### Purpose

**Verify that there are no missing CI connections.**

Pay attention to CIs with no connectors from it to other CIs. For example, in the following figure, the Web Server on win2K12 CI is missing connections.

![Diagram showing missing connections](image)

If connections and CIs to which they lead are missing, the map does not reflect the real state of the application service and its operation. Inaccurate data can also be transferred to Event Management, causing imprecise monitoring.

### Action

**Check that the connections between CIs are correct.**

View CI connection attributes in an application service map.
**Purpose**

Check that clusters are reflected correctly.

**Action**

Click the plus (+) icon next to a CI.

There are two types of clusters:

**Application cluster**

In an application cluster, two or more applications or devices are configured to work together and serve the same purpose. Typically, the clusters provide high availability or load balancing. For example, web servers working behind a load balancer.

This type of cluster appears as a stack of CIs with a label showing the number of CIs in this cluster with the multiplication sign.

Application clusters can have a cluster within a cluster, but never more than two levels:
<table>
<thead>
<tr>
<th>Purpose</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check that inclusions are reflected correctly.</td>
<td>Click the plus (+) icon next to a CI. In an inclusion, a server hosts applications that are treated as independent objects. For example, IIS Virtual Directory can run on a Windows Service as its host.</td>
</tr>
</tbody>
</table>

4. To notify the service mapping admin of a necessary change or fix, perform the following steps:
   
a) Right-click the CI or connection to which you want to add your comment, and select **Submit Reject Message**. Or
   
   To add a general comment for the application service, click **Actions** on the **Add Reject Message** tab under the map.

   b) In the Add Reject Message window, enter your comment, and click **Add**.
   
   The new comment appears on the **Discovery Messages** tab under the map. If you added the comment to a specific CI or connection, the map shows a warning icon (⚠️) next to them. The application service status changes to Rejected.

   c) If necessary, add more comments.

5. If you are satisfied with the results, approve this application service by clicking **Actions**, and then **Approve service**.

   The application service status changes to Approved and it appears in the list of completed application services on the **Home** page. The operational status changes to Operational. From this point, other applications can use data collected and organized by Service Mapping.

**Fine-tune application services to implement owner requests**

As an administrator, you tweak and fine-tune maps based on comments from the application service owner.

Role required: sm_admin

Reviewing and implementing owner requests is part of the **review and approval process**.

To see documentation for another review phase, click the relevant box in the diagram.
1. **Send the application service for review**

2. **Review the application service and request fixes**

3. **Resend the application services for review**

4. **Review and approve the application service**

1. Navigate to the relevant application service:
   a) Click **Approve** on the **Home** page.
      The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.
   b) Click **Fix Rejected**.
   c) Right-click the required application service and select **Open Service Form**.
      The form for this application service opens.

2. **Click Reject Messages** on the left.
   The owner comments are displayed under the **Service Discovery Messages** tab.
3. Review the reject messages to see comments that the application service owner left.
4. Click **View Map**.
5. Implement changes requested by the application service owner.
6. If necessary, resend the application service for review.

**Manually add CIs to an application service**

Add configuration items (CIs) to manually created application services or to services discovered by Service Mapping.

- Verify that the CI type for the CI that you are planning to add, exists. If necessary, create the CI type as described in *Create CI types for Service Mapping and Discovery*.
- Add CIs to the CMDB for the device or application that you want to add, if necessary. See *Populate the CMDB* for more information.

**Role required:** **app_service_admin** or **sm_admin**

You can populate an application service that was created manually, by manually adding the CIs comprising it. Adding a CI to an application service requires creating a relationship between the new CI and a CI in the application service.

Information about the CI in application service, to which you are connecting a new CI, is updated in the CMDB. This information includes the type of the relationship between the CIs. If other application services use the same applicative flow, the CMDB recognizes it and adds the CI you added manually to these application services by analogy. For example, you manually added an IBM WebSphere Message Broker to an IBM WebSphere HTTP Listener in the Bank Customer Portal service. The system also adds this IBM WebSphere Message Broker to the same HTTP Listener in the Bank Internal Portal, because it uses this HTTP Listener. The same logic applies when you remove a CI you added manually: The system removes it from all application services where you either manually added it or the system added it by analogy.

All connections created between CIs in an application service are of the **Depends on:** : **Used by** relationship type. You can modify this default relationship type by changing the value of the *sa.it_service.manual_ci_rel_type* property.

You can manually connect a CI only to actual CIs existing in the CMDB, not to a visualization of other items on the map such as clusters or boundaries. Also, it is prohibited to add CIs of these CI types to an application service:

- **cmdb_ci_endpoint**
- **cmdb_ci_translation_rule**
- **cmdb_ci_config_file**
- **cmdb_ci_qualifier**
- **cmdb_ci_application_cluster**

If necessary, you can prevent CIs of other CI types from being added to application services by modifying the *sa.mapping.user.manual.citype.blacklist* property.

In environments with domain separation, only CIs belonging to the same domain as the application service are added to the application service. If there is a domain hierarchy, CIs must belong to the same child domain as the application service.

If working with an application service discovered by Service Mapping, manually add a CI:

- To indicate that an application service contains a device or application, which Service Mapping cannot discover. For example, add an A/C unit to the Production Floor service.
- To add a temporary placeholder for a CI, which Service Mapping did not discover. In this case you are planning to perform necessary troubleshooting to ensure that Service Mapping discovers this CI in the future. For example, add an IBM WebSphere Message Broker to the Bank Customer Portal service.
• To create an application service that combines entry points and CIs automatically discovered by Service Mapping with entry points and CIs from the CMDB. After you manually add an entry point, you can update the application service with CIs from the CMDB based on the relationships defined there.

For additional information related to Service Mapping, see Discovery patterns used by ITOM Visibility and Enable traffic-based discovery for CI types or specific CIs.

1. Navigate to Configuration > Application Services > Application Services.
2. Alternatively, if Service Mapping is deployed, navigate to Service Mapping > Services > Application Services.
3. Open the relevant application service map:
   a) Click View map next to the relevant application service.
   b) If needed, click Edit to ensure that the map is in Edit mode.

   If Service Mapping is deployed, then in Edit mode, the Discovery Messages section appears below the map.

4. To connect a CI to another CI on the map, right-click the CI to which you want to connect the new CI, and then select Add a CI.
5. In the Add A CI dialog box, specify the CI to add:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Select the CI type (CI class) for the CI you are adding. Every CI belongs to a CI type which contains a set of attributes configured for this kind of CI, for example, cmdb_ci_appl for applications.</td>
</tr>
<tr>
<td>CI Name</td>
<td>Select the CI from the list of CIs of the selected CI type.</td>
</tr>
</tbody>
</table>

   The CI type list includes only allowed CI types. For example, you cannot add an application cluster.

6. Click Submit.

   The manually added CI appears on the map.

   **Note:** When you manually add a CI, which is an application, as a child to a service that already includes its parent application CI, the newly added child application CI is hidden inside the inclusion. Click the plus (+) symbol next to the parent application CI to see the child application CI.

7. Optional: (If Service Mapping is activated) To add a discoverable outgoing connection for the manually added CI:
   a) Right-click the manually added CI.
   b) Select Manually add a connection.

   **Note:** If you do not see the Manually add a connection option in the right-click menu, check that you are logged in with the user that belongs to the same domain as the application service.

   c) Configure attributes for the entry point as described in Entry point attributes.
   d) Click Submit.
Discovery and Service Mapping attempt to discover this CI. If successful, the CI appears on the map. Otherwise, a warning icon (⚠️) appears.

8. Optional: (If Service Mapping is activated) If you want Service Mapping to automatically discover a CI, which you previously added manually:
   a) Customize the relevant pattern or fine-tune traffic-based discovery to enable Service Mapping to discover the CI.
   b) Navigate to the relevant application service map.
   c) Click **Run discovery**.
   d) After the discovery process finishes, verify that Service Mapping discovered the CI by checking the connector leading to the CI. If Service Mapping discovered the CI, then two connectors, a manual and automatically discovered, appear for the CI.

   ![Diagram of service map with manual and automatically discovered connectors]

   e) Right-click the CI you added manually.
      In the example, it is IBM WebSphere Message Broker.
   f) Select **Remove manually added CI**.
The map shows the CI with only one connector leading to it. If this CI had any manually added connections, they are removed together with the manually added CI.

Remove CIs not belonging to application services
If Service Mapping erroneously mapped some CIs as part of an application service, remove these CIs for noise reduction. It is important to remove unnecessary CIs from the map, because they can generate irrelevant alerts in Event Management. For example, when creating an application service for a web portal, Service Mapping may automatically discover a connection to external services, such as PayPal service, that do not belong to the service.

Role required: sm_admin
Removing CIs from the map also removes all their connections. For example, if you remove a CI of the second tier on the map, all CIs of the third tier connected to it through outgoing connections, get removed too.

Depending on why Service Mapping mapped an unwanted CI, you can remove it by the following methods:

- Add mapping boundaries to remove the unnecessary CIs. A boundary makes Service Mapping stop discovery from this point and not follow outgoing connections. Use this method to remove CIs that Service Mapping discovered using patterns or using traffic-based discovery.
- Tweak traffic-based discovery. Service Mapping uses traffic-based connections to collect network statistics and perform traffic-based discovery. This discovery method may clutter an application service with irrelevant CIs. Service Mapping uses an algorithm to reduce the number of erroneously mapped CI. You can further adjust traffic-based discovery to remove unwanted CIs.

1. Navigate to Service Mapping > Services > Application services.
2. Click View map next to the relevant application service.
3. To remove CIs, right-click the connector leading to the CI you want to remove, and select Mark boundary.

The CI is marked as boundary on the map ( ).
4. To remove CIs discovered using traffic-based discovery, perform the following steps:
   a) Right-click the CI on the map.

   Note: You can see traffic-based connections for a CI, even if you disabled the traffic-based discovery for an application service the CI is part of.
   b) Select Show traffic based connections.

   The Traffic Based Connections List opens displaying the following information:

   Traffic Based Connections List

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>The IP address of the application connected to the selected CI.</td>
</tr>
<tr>
<td>Port</td>
<td>The port on the selected CI that is used to communicate to the other application.</td>
</tr>
<tr>
<td>Process</td>
<td>The ID of the process in the selected CI.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Already on map      | • Yes — if this connection shows on the map.  
• No — if this connection is not part of the application service and not on the map.   |
| System decision     | The setting defines if Service Mapping keeps the discovered traffic-based connection or removes it. The value comes from the algorithm that Service Mapping uses. |
| User decision       | (Optional) The setting overrides the System decision setting, which defines if Service Mapping keeps the discovered traffic-based connection or removes it. For example, if the System decision setting for a connection is Keep, and you want to remove this connection, select Remove. |

**Note:** There may be a case when you can see traffic-based connections on the map, but the Traffic Based Connections list does not display them. It happens for the connections that have been removed from the TCP Connection (cmdb_tcp) table less than three days.

c) In the **Traffic Based Connections List**, identify the connection used to discover the unwanted CI.
d) Select **Remove** from the **User decision** field for this connection.
e) Click **Close**.

**Transfer a map segment into another application service**

You can remove a branch of a service and place it into another application service, either brand new or existing. The original map shows the link to the other application service instead of the map segment.

**Role required:** sm_admin

Transfer map segments when you need to split large services or when you want to organize services differently from the initial mapping result.

When you transfer a map segment into another application service, you create a link between the original application service and the application service into which you planted the map segment. The connection to the top CI in the segment becomes an entry point when you transfer it into another, existing or new, application service.

When you transfer a map segment into another application service, the information about the connection to the top CI in the transferred segment, is updated in the CMDB. If other application services use the same ingoing connection (endpoint), the CMDB recognizes it and replaces map segments coming out of the same connection with connected service CIs by analogy.

You cannot insert links to multiple application services from the same connection.
If Event Management is activated in your organization, you can monitor impact of the transferred segment in the original application service containing the link.

1. Open the application service map containing the segment that you want to transfer:
   a) Navigate to Service Mapping > Services > Application Services.
   b) Click View map next to the relevant application service.
   c) Optional: Click Edit to be in Edit mode.
   
   ![Application Service Map](image)

   d) If the Host view is on, switch it off by clicking More Options and clicking the Display in Host View toggle.
2. Identify the segment of the map that you want to transfer.
3. Right-click the connection leading to the CI at the top of this segment.

4. To transfer the segment into an existing application service map and connect to it:
   a) Select **Connect to existing service**.
b) In the Add To Existing Discovered Service window, select the application service to which you want to transfer the segment.

c) Click Choose.

The map displays the connected application service icon instead of the transferred segment. The application service map to which you transferred the segment, shows it under the newly added entry point. For example, the Tomcat application service map contains the Trade connected service CI. The Trade application service map shows the segment transferred from the Tomcat application service.

5. To transfer the segment into a new application service map and connect to it:

   a) Select Create new service from here.

   b) In the Create New Discovered Service window, define attributes for the application service that you want to create for this segment:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter the application service name. This name must be unique. Use self-explanatory names such as mailing service or printing service.</td>
</tr>
<tr>
<td>Group</td>
<td>(Optional) Restrict access to an application service by adding it to an application service group. Users must then have the service group role to access the application service.</td>
</tr>
<tr>
<td>Criticality</td>
<td>(Optional) Select the option that reflects how important this application service is to your organization operation. For more information about application service criticality, see Define criticality for application services.</td>
</tr>
</tbody>
</table>
c) Click **Create**.

The map displays the connected application service icon instead of the transferred segment. The new application service appears in the list of application services and contains the transferred segment of the map. For example, the Tomcat application service includes the TP rooms connected service CI. The TP rooms application service map contains the segment transferred from the Tomcat application service.

To revert this operation:

1. On the application service map containing the connected service CI, click the connection leading to it.
2. In the Properties pane, note the URL attribute of the connection.
3. Double-click the connected service CI.
   The application service map for the CI containing the transferred segment opens.

4. Click **Service Map Form**.

5. Under Entry Points on the left, locate the entry point with the same URL you took note of.

6. Click **Delete** on the entry point tab, and click **Remove** to confirm.
   This entry point disappears from this application service form.
   The transferred segment disappears from the map of this application service.

7. Navigate to the map that contained the connected service CI.
   This map displays a boundary instead of the connected service CI.

8. Click the boundary CI and click **Unmark boundary**.
   The map is refreshed and displays the segment of the application service.

Service maps contain no indication of transferred services being linked to their original services.
You may need to know if a service is a part of another, larger service when dealing with changes or incidents.
Use the updateset to add the ability to view the list of all application services that contain a certain service as a link, as described in **KB0748865: Viewing list of application services that contain a certain service as a link**.

**Link application services**
You can manually add to one application service a reference to another application service.
Ensure that you know the name of the application service that you want to add a reference to.
Role required: app_service_admin or sm_admin

An application service can be part of another application service, in which case you can link one to the other. To link, edit an application service, and add a reference to another application service as an outgoing connection of the relevant CI inside the other application service.

When you link an application service to another application service, the information about the CI, to which you linked the service, is updated in the CMDB. The CMDB recognizes other application services that use the same applicative flow, and adds the application service you linked to these application services by analogy. The same logic applies when you remove a linked application service: The system removes it from all application services where you either manually linked this service or the system linked this service by analogy.

When using Service Mapping, you may want to link application services to create:

- A link between two discovered services.
- A placeholder for a map branch that Service Mapping failed to discover. If you modify or add a pattern that can discover the configuration item (CI) serving as an entry point for the linked application service, Service Mapping can discover this linked service.
- An indication that an application service contains a branch, which Service Mapping cannot discover.

You can link application services to one another to create dependencies for impact monitoring in Event Management.

1. Navigate to Configuration > Application Services > Application Services.
2. Alternatively, if Service Mapping is deployed, navigate to Service Mapping > Services > Application Services.
3. Click **View map** next to the application service which you want to link to another application service.

4. Right-click the CI to which you want to link an application service as a reference.

5. Select **Add A CI**.

6. In the Add a CI dialog box, select the application service you want to add as a reference:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Select <em>Application service</em> from the list.</td>
</tr>
<tr>
<td>CI Name</td>
<td>Select the name of the application service that you want to link as a reference.</td>
</tr>
</tbody>
</table>

7. Click **Submit**.
   
   The icon for the linked service appears on the map.

Navigate to the linked application service, and double-click its icon on the map. The map window shows the linked application service.

Service maps contain no indication of transferred services being linked to their original services. You may need to know if a service is a part of another, larger service when dealing with changes or incidents. Use the updaterset to add the ability to view the list of all application services that contain a certain service as a link, as described in [KB0748865: Viewing list of application services that contain a certain service as a link](https://servicenow.com/knowledgebase/docs/KB0748865).

**Discover CIs identified as generic applications**

Service Mapping identifies application configuration items (CIs) that it failed to properly discover as generic applications. Correctly identify generic applications by creating a simplified discovery pattern from a business service map, rather than creating a fully functional pattern from scratch using the Pattern Designer.

Make sure that you understand what **discovery process classification** is.

Request information about the CI discovered as generic application from the business service owner.

Role required: pd_admin

ServiceNow applications refer to devices and applications that comprise an application service as configuration items (CIs).

As part of creating a simplified pattern for complete and proper application CI discovery, you create these necessary articles:

**CI type**

All applications and hosts in your organization must have a corresponding configuration item (CI) type, which is necessary for discovering and processing applications and hosts correctly. A CI type (or class) contains several important definitions that apply to all CIs belonging to it. Preconfigured CI types form a hierarchy where child CI types derive attributes from their parents.

Define the CI type name and label for Service Mapping to create a child CI type for the application (cmdb_ci_appl) CI type. For more information about CI types, see [Create CI types for Service Mapping and Discovery](https://servicenow.com/knowledgebase/docs/Create_CI_types_for_Service_Mapping_and_Discovery).

**Process classification**

Service Mapping relies on process information collected and classified by Discovery to run relevant patterns to discover an application CI. During the horizontal discovery process, Discovery checks if the attributes of the processes running by the application match classification criteria you define. When a match is found, Discovery uses the existing Application
Rule identifier to check if the application CI is new or if it already exists in the CMDB. Then Discovery either updates an existing record or creates a new record for this application CI in the CMDB.

Prior to starting the top-down discovery of an application CI, Service Mapping checks its process classification in the CMDB. It then runs only those patterns that are relevant for this application CI.

Create process classification by defining the classification criteria based on preconfigured process attributes. The new process classification is added to the Process Classification (discovery_classy_proc) table. To learn about process classification, refer to Discovery classifiers.

**Pattern**

A pattern is a sequence of steps whose purpose is to detect attributes of a CI and its outbound connections. When you create a simplified pattern for a generic application, Service Mapping assigns the new simplified pattern to the CI type you define as part of this process. The default name for the pattern starts with “ga_”.

**Association of an icon with this CI type**

Each CI type is associated with an icon. When a CI is discovered and classified as belonging to a CI type, the map displays the associated icon to visualize this CI.

You cannot create simplified patterns from generic applications that are based on multiple processes. In this case, you have to either create a fully functional pattern using the Pattern Designer or create a new CI type.

1. Navigate to Service Mapping > Services > Application Services.
2. Click View Map next to the relevant application service.
3. Optional: Click Edit to be in Edit mode.
4. On the map, right-click the relevant generic application CI. The **CI Type** attribute in the **Application** pane is **Generic Application**. In the base system, the map displays generic application CIs with the gear icon.
5. Select **Create pattern for this generic application**.

6. Define the CI type for the application.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Label</strong></td>
<td>CI type name. For example, Citrix Licensing Server.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>The table name containing data for this CI type. For example, cmdb_ci_app_license_server. Use a name that is consistent with existing CI type names. Typically, application CI type names start from &quot;cmdb_ci_app&quot;.</td>
</tr>
<tr>
<td><strong>Icon</strong></td>
<td>Select the name of the existing icon to represent this application on maps.</td>
</tr>
</tbody>
</table>

7. Create the process classification for this CI by defining classification criteria for identifying the process that the generic application is running.

   In the base system, there are two conditions based on the running process name and the running process command. The values are filled in by the latest process discovery.

   a) Select the CI attribute from the list.

      The attributes in the list are from the Running Process (cmdb_running_process) table.

   b) Select an operator from the list.

      If you select **Is Empty**, the second field is rendered irrelevant and disappears.

   c) Specify the value to match.

      The value field displays a dynamically generated suggestion from the latest process discovery.
d) To add a dependent condition, click OR or AND.

**Note:** If the value in the first condition is not unique like Java.exe, use additional conditions to narrow down the results.

For example, to discover a Citrix Licensing server, use the following classification criteria: the process name `Imadmin.exe` with the command `C:\Program Files (x86)\Citrix\Licensing\LS\Imadmin.exe`.

e) To add a top-level condition, click **New Criteria**.

8. **Click Create.**
   
   Once Service Mapping finished creating the CI type, the pattern, and the process classification, it resumes discovery of the CI initially identified as a generic application.

9. Verify that Service Mapping discovered the CI as you expected.
Notice that Service Mapping uses the name of the process for the CI label on the map.

If Service Mapping still has not discovered the CI as you expected, fine-tune the match and classification criteria for the process classifier and configure advanced classification parameters.

If you are familiar with modifying patterns in the Pattern Designer, you can also enrich the simplified pattern by defining operations that discover more attributes for the CI.

You may configure Discovery to use a different identifier. For more information, see Discovery identifiers.

**Application service completion**

After an application service is reviewed and approved, you can define attributes that enhance its discovery, reflect its importance, and control access to it.

There are several configurations you must perform to complete definition of an application service.
Define criticality for application services

For each application service, select a criticality level that reflects how important it is to your organization operations.

Determine how important the application service is to your organization.

Role required: sm_admin
Typically, you might assign high criticality to application services, which impact your customer base or sales revenue, while giving low criticality to internal application services causing minor inconvenience to organization employees. For example, you could set high criticality for an application service that supports sales functionality using the organization web site. An application service that provides internal printing for the organization employees could receive low criticality.

Criticality determines the prominence of an application service on the Event Management dashboard. You can also use a criticality value to define recovery strategies.

Example of an application service criticality on the Event Management dashboard

1. In the classic UI, perform the following steps:
   a) Navigate to **Service Mapping > Home**.
      The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.
   b) Click **View Your Services** on the **Completed** tile.
c) Select the relevant application service.
d) Click **Additional Info**.
e) Select the appropriate option from the **Business criticality** list.
f) Click **Update**.

2. Alternatively, perform the following steps in the Agent Workspace UI:
   a) Navigate to **Agent Workspace > Agent Workspace Home**.
b) Click **Lists** in the left pane.
c) Navigate to **Services > All Services**.
The list of services displays services in Operational status.
d) Select the relevant application service.
e) Click the **Open Preview** icon.

The Details pane opens on the right.
f) Select the appropriate option from the **Business criticality** list.
g) Click **Update**.

---

### Enable traffic-based discovery for an application service

Make Service Mapping use traffic-based connections to discover CIs making up an application service.

Role required: sm_admin
Service Mapping uses traffic-based connections to collect network statistics and perform traffic-based discovery. Not only can Service Mapping discover CIs using patterns, it can also discover them by following traffic connections between CIs. This method is referred to as traffic-based discovery. For more information about how Service Mapping performs traffic-based discovery, see Traffic-based discovery in Service Mapping.

1. In the classic UI, perform the following steps:
   a) Navigate to Service Mapping > Home.
      The Home page displays only information on application services that Service Mapping can discover or already discovered. The Home page does not display information on application services that are created manually or using the API.
   b) Click View Your Services on the Completed tile.
   c) Select the relevant application service.
   d) Click Additional Info.
   e) Click the Traffic based discovery check box.
   f) Click Update.

2. Alternatively, perform the following steps in the Agent Workspace UI:
   a) Navigate to Agent Workspace > Agent Workspace Home.
   b) Click Lists in the left pane.
   c) Navigate to Services > All Services.
      The list of services displays services in Operational status.
   d) Select the relevant application service.
   e) Click the Open Preview icon.
The Details pane opens on the right.

f) Click the Traffic based discovery check box in the Details pane.

g) Click Update.

Group application services

Organize application services by groups to perform actions simultaneously on multiple services, and to control user access to services. You can use Event Management to track service health by service groups.

Role required: sm_admin or app_service_admin

Typically, enterprises have hundreds of services which makes it impractical to manage them individually. Service groups can make service lists much shorter and easier to manage, especially in large organizations or service providers.

How you group application services depends on the user and on service provisioning policies in your enterprise. The relation between application services in groups is purely logical and the same application service can belong to multiple groups. For example, the Mobile service can be part of the following service groups: sales, Beijing, and telephony.

Example of an application service belonging to different groups

You can embed a service group within another service group to create a hierarchy of service groups. If users have access to a parent service group, they automatically have access to all its child groups. By default, all services are assigned to the All service group that lets all users view and manage application services. When you assign a role to a service group, the users with this role can access only application services in this service group. To enable users with this role to access other services, assign this role to the respective service group.
If Service Mapping is activated, service groups can contain a mixture of manually created application services and application services discovered by Service Mapping.

1. Navigate to Configuration > Application Services > Service Groups. Or, if Service Mapping is activated, navigate to Service Mapping > Services > Service Groups.
2. Click New.
3. Enter the name of the new application service group in the Name field.
4. To embed this group in another group, enter the name of the other group in the Parent Group field.
5. Right-click the form header and click Save.
6. Add an application service to the newly created service group.
   a) In the Service Group Members section, click New.
   b) In the Name field, enter the name of the application service. If you are using Event Management, you can also enter an alert group name.
   c) Click Submit.
7. Alternatively, add an application service to a group from the application service form.
   a) Navigate to Configuration > Application Services > Application Services.
   b) Select the application service you want to add to a service group.
   c) In the Service Group Members section, double-click Insert a new row.
   d) Enter the name of the service group to which you want to add the selected application service.
   e) Click the OK icon (✔).
   f) Click Update.

Control user access to application services

Assign user roles to service groups to grant users access to application services in your organization. Your organization may restrict access to some services for security or secrecy reasons.

Make sure that you have performed the user provisioning tasks for the users you want to grant access:

1. Add users to user groups.
2. Create new roles.
3. Assign roles to users or user groups.

Also, make sure that you have created service groups as described in Group application services.

Role required: app_service_admin or sm_admin

In the base system, the following roles provide access to application services:

app_service_admin

Creates and modifies application services, creates service groups, views, and edits application service maps.

app_service_user
Views application service maps. The `itil` role that serves as the basic helpdesk technician role contains the `app_service_user` role.

Service Mapping provides these preconfigured roles:

**sm_admin**
Sets up the Service Mapping application. Maps, fixes, and maintains application services. Also performs advanced configuration and customization of the product. Assign this role to application administrators.

**sm_user**
Views maps for operational application services to plan change or migration, as well as analyze the continuity and availability of services. Assign this role to application users.

**sm_app_owner**
Provides information necessary for successful mapping of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. Assign the `sm_app_owner` role to users who own application services and are familiar with the infrastructure and applications that make up the services.

Event Management provides these preconfigured roles:

**evt_mgmt_admin**
Has read and write access to all Event Management features to configure Event Management.

**evt_mgmt_operator**
In addition to the `evt_mgmt_user` permissions, can also activate operations on alerts such as acknowledge, close, open incident, and run remediations.

**evt_mgmt_user**
Has read access to all Event Management features. Has write access to alerts to manage the alert life. Has the `itil` role to be able to manage incidents that are created from alerts.

**evt_mgmt_integration**
Has create access to the Event (em_event) and Registered Nodes (em_registered_nodes) tables to integrate with external event sources.

Typically, enterprises have hundreds of services which makes it impractical to manage them individually. Service groups can make service lists much shorter and easier to manage, especially in large organizations or service providers. In a hierarchy of service groups, access to a parent service group automatically grants access to all the child service groups.

Users inherit permissions from roles that are assigned to them. You can assign some roles directly to service groups to allow all users with this role to access all application services belonging to this group. However, most enterprises choose to organize their roles as a hierarchy. It helps to manage roles across multiple ServiceNow applications. For example, the Service Mapping administrator (`sm_admin`) can be part of a broader administrator role like administrator (`admin`). You can add users to user groups and then assign roles to the user groups to give permissions of this role simultaneously to all the group users.
Assigning a role to an application service group

By default, all services are assigned to the **All** service group that lets all users view and manage application services. When you assign a role to a service group, the users with this role can access only application services in this service group. To enable users with this role to access other services, assign this role to the respective service group.

**Note:** Some references in the user interface to business services are actually references to application services.

1. Navigate to either of the following:
   - **Configuration > Application Services > Service Group Responsibilities.**
   - If Service Mapping is activated: **Service Mapping > Services > Service Group Responsibilities.**
   - If Event Management is activated: **Event Management > Services > Service Group Responsibilities.**

2. Click **New** and fill out the Business Service Group Responsibilities form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Service Group</td>
<td>Service group to which you want to assign a role.</td>
</tr>
<tr>
<td>Role</td>
<td>Role you want to assign to the selected service group. For example, financial_services_admin.</td>
</tr>
</tbody>
</table>
3. Click Submit.

To manage access to services that contain sensitive financial information in your organization:

1. Organize the services into the Financial Services group.
2. Create a new user role, financial services administrator (financial_services_admin) role, that contains the (app_service_it) role.
3. Assign the Financial Services administrator role to the Financial Services group.

As a result, only users with the Financial Services administrator role can access application services belonging to the Financial Services group.

Schedule a top-down discovery by Service Mapping

After Service Mapping discovers configuration items (CIs) belonging to your application service for the first time, it then rediscovers CIs to find changes and updates. Create or modify discovery schedules to control how often Service Mapping rediscovers services or CIs.

Define your application services prior to defining discovery schedules for discovering CIs which are part of these services.

When creating a schedule for CI types or specific CIs, make sure that at least one application service containing the relevant CI is in the operational state.

Role required: sm_admin

When you define a new application service, Service Mapping performs discovery of all CIs that participate in this application service and creates its map. After the initial mapping is complete, Service Mapping regenerates application service maps regularly by rediscovering CIs making up an application service.

By default, Service Mapping is preconfigured with these generic schedules:

**All Applications**

This schedule triggers the top-down discovery of all CIs of the application type (cmdb_ci_appl).

**Load Balancer Service**

This schedule starts the top-down discovery of all CIs of the load balancer service type (cmdb_ci_lb_service)

The generic schedules trigger discovery of all applications and load balancer services in your organization once a day. Typically, these preconfigured schedules are enough to update information for application services. However, if your organization needs to discover specific CIs or services more frequently, you can create the following custom discovery schedules:
Schedule types for top-down discovery

<table>
<thead>
<tr>
<th>Schedule type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Services</td>
<td>Service Mapping discovers all CIs belonging to certain services that answer filtering criteria.</td>
<td>In your organization, some application services are more critical than others and it is important for you to rediscover such services with high criticality more frequently. Create a custom discovery schedule to discover all services with the business criticality value set to <strong>most critical</strong>. Alternatively, certain services in your organization are undergoing changes more often than once a day. You may want to create custom schedules to rediscover such services more frequently. You can also create a schedule of this type to discover application services in a sub-production instance, where services are not in the <strong>operational</strong> state yet.</td>
</tr>
<tr>
<td>Services belonging to a group</td>
<td>Service Mapping discovers all CIs belonging to services assigned to groups that answer filtering criteria. Service Mapping discovers only services located directly inside the group that answers defined filtering criteria. Service Mapping does not discover services inside embedded groups. Notice that you can define schedules based on service groups, even if your role does not have access to these service groups.</td>
<td>In a distributed organization with offices in several geographic locations, you may want to create a custom discovery schedule to rediscover services in one of these locations. For example, one schedule can trigger rediscovery of all services relevant for the EMEA site at 7am GMT, while another schedule can start rediscovery of services for the US headquarters at 7am PT.</td>
</tr>
<tr>
<td>Schedule type</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td>---------</td>
</tr>
<tr>
<td>For CI types</td>
<td>Service Mapping discovers all application CIs belonging to this application type. Some CI types are prone to more frequent changes and updates than others, so you can manage the load by adjusting the discovery schedule to match the nature of each CI type. When you define discovery schedules based on application CI types, several schedules may apply to the same CI. To avoid discovering the same CIs more than once, the most specific schedule always has precedence. For example, if you create separate discovery schedules for a parent CI type and its child CI type, CIs belonging to the child CI type are discovered using its dedicated schedule. At the same time, if there is no schedule for a child CI type, the parent CI type schedule is used to discover the child CIs. This schedule discovers only CIs belonging to application services in the operational state.</td>
<td>If you modify a certain application more often than the rest of your applications, you may want to discover it more frequently than other CIs. In that case, create a custom discovery schedule for such an application CI type.</td>
</tr>
<tr>
<td>Specific CIs</td>
<td>Service Mapping discovers only one CI that you specified for this schedule. If you define a discovery schedule for a specific CI as well as a schedule for the CI type to which this CI belongs, Service Mapping uses the schedule for this specific CI, and not the generic schedule for its CI type. This schedule discovers only CIs belonging to application services in the operational state.</td>
<td>For rare cases of rediscovering a specific CI that causes discovery errors frequently.</td>
</tr>
</tbody>
</table>

Having customized discovery schedules also allows you to avoid redundant stress on your infrastructure.

Part of discovering an application CI is identifying its host. Service Mapping checks if the device hosting this application CI exists in the CMDB. For a load balancer service, Service Mapping checks if the load balancing device hosting this load balancer service exists in the CMDB. If not, Service Mapping triggers Discovery to perform the horizontal discovery. As a result, Discovery performs host detection and updates the information on hosts in the CMDB. If necessary, you can manage the schedules that trigger horizontal discovery as described in:

- **Schedule a horizontal discovery** — for standard horizontal discovery.
- **Create a schedule for standard serverless Discovery** — for discovering CIs using a protocol such as REST, without using the IP address of any devices or hosts.
- **Create a schedule for host-based serverless Discovery** — for discovering CIs that reside on an already discovered host using this host as a Proxy server.

1. Navigate to Service Mapping > Administration > Discovery Schedules.
2. To create a custom discovery schedule, click New.
Or

To customize the generic schedule for all application CIs, click All Applications or Load Balancer Service.

The Discovery Schedule page opens.

3. In fresh install or reset instances, fill in the fields as follows:

**The Discovery Schedule form in fresh install or reset ServiceNow deployments**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>For a new specific schedule, enter a unique and descriptive name.</td>
</tr>
<tr>
<td>Discover</td>
<td>By default, the value for this field is Service, indicating that you are creating a top-down discovery schedule. You cannot change this value.</td>
</tr>
<tr>
<td>Discover by</td>
<td>For a new specific schedule, select CI Type.</td>
</tr>
<tr>
<td>CI type</td>
<td>This field is available when CI type is selected from Select Discovery. For a new CI type-based schedule, select the relevant CI type from the list.</td>
</tr>
<tr>
<td>CI</td>
<td>This field is available when Specific CI is selected from Select Discovery. For a new schedule for a specific CI, select the relevant CI from the list.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter Services to Discover</td>
<td>This field is available when Service Attributes is selected from Select Discovery. Use the filter to create the filtering criteria for services. Service Mapping discovers only services that answer the defined criteria.</td>
</tr>
<tr>
<td>Filter Groups to Discover</td>
<td>This field is available when Service Group Attributes is selected from Select Discovery. Use the filter to create the filtering criteria for service groups. Service Mapping discovers only services belonging to the groups that answer the defined criteria.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to enable this schedule. If you clear the check box, the schedule is disabled, but you can still run a discovery manually from this form, using the configured values.</td>
</tr>
<tr>
<td>Max run time</td>
<td>Set a time limit for running this schedule. When the configured time elapses, the remaining tasks for the discovery are canceled, even if the scan is not complete. Use this field to limit system load to a desirable time window. If no value is entered in this field, this schedule runs until complete.</td>
</tr>
</tbody>
</table>

4. In upgraded deployments, fill in the fields as follows:

#### The Discovery Schedule form in upgraded ServiceNow deployments

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>For a new specific schedule, enter a unique and descriptive name.</td>
</tr>
<tr>
<td>Discover</td>
<td>By default, the value for this field is Service, indicating that you are creating a top-down discovery schedule. You cannot change this value.</td>
</tr>
<tr>
<td>Service discovery</td>
<td>For a new specific schedule, select a schedule type as described in Schedule types for top-down discovery.</td>
</tr>
<tr>
<td>CI type</td>
<td>This field is available when CI Type is selected from Select Discovery. For a new CI type-based schedule, select the relevant CI type from the list.</td>
</tr>
<tr>
<td>CI</td>
<td>This field is available when Specific CI is selected from Select Discovery. Select the relevant CI from the list to create a discovery schedule for this CI.</td>
</tr>
<tr>
<td>Filter Services to Discover</td>
<td>This field is available when Service Attributes is selected from Select Discovery. Use the filter to create the filtering criteria for services. Service Mapping discovers only services that answer the defined criteria.</td>
</tr>
</tbody>
</table>
5. Optional: For a discovery schedule based on service attributes, click View Selected Services to preview the list of services that answer the filtering criteria you entered. Click Close.

6. Optional: For a discovery schedule based on service group attributes, click View Selected Groups to preview the list of groups that answer the filtering criteria you entered. Click Close.

7. Define the rediscovery frequency by selecting the relevant option from the Run list:

### Discovery schedule run options

<table>
<thead>
<tr>
<th>Run option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>On Demand</td>
<td>Does not run on a schedule. Click Discover now to run Discovery. This is the default option.</td>
</tr>
<tr>
<td>Daily</td>
<td>Runs every day. Use the Time field to specify the time of day.</td>
</tr>
<tr>
<td>Weekly</td>
<td>Runs on one designated day of each week. Use the Time field to specify the time of day.</td>
</tr>
<tr>
<td>Monthly</td>
<td>Runs on one designated day of each month. Use the Day field to select the designated day. Use the Time field to select the time of day. If the designated day does not occur in the month, the schedule does not run in February.</td>
</tr>
<tr>
<td>Periodically</td>
<td>Runs every designated period of time. Use the Repeat Interval field to define the period of time in days, hours, minutes, and seconds. The first Discovery runs at the point in time defined in the Starting field. Repeal Interval period passes.</td>
</tr>
<tr>
<td>Once</td>
<td>Run one time as designated by the date and time defined in the Starting field.</td>
</tr>
<tr>
<td>Weekdays</td>
<td>Runs every Monday, Tuesday, Wednesday, Thursday, and Friday. Use the Time field.</td>
</tr>
<tr>
<td>Weekends</td>
<td>Runs every Saturday and Sunday. Use the Time field to select the time of day.</td>
</tr>
<tr>
<td>Month Last Day</td>
<td>Run the last day of every month. Use the Time field to select the time of day.</td>
</tr>
<tr>
<td>Calendar Quarter End</td>
<td>Runs on March 31, June 30, September 30, and December 31. Use the Time field to select the time of day.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
8. Click Submit.

Service definition transfer from one instance to another

Save time and effort by exporting definitions of application services from the source instance and importing these definitions into the target instance. Use this method to copy definitions of discovered application services only, not manually created application services.

Typically, you maintain two ServiceNow instances in your organization: for production and for development. You perform the initial mapping of application services in your organization on the development instance, which you use for testing. The results of initial mapping often require some basic troubleshooting or fine-tuning. Once you finish mapping application services on your source instance, transfer service definitions into the target (production) instance.

Notice that you cannot export the actual application services, only service definitions. Mapping is a dynamic process, and the mapping result reflects the actual infrastructure and its configuration. Therefore, the services you exported from one instance and imported into another may look different.

By default, you export the following service definitions:

- Service name
- Traffic-based discovery usage
- Entry point
- Business criticality
- Group to which the service belongs
  
  If this service group is part of a service group hierarchy, all parent service groups in this hierarchy are exported as well.
- Comments

The system uses the Sys ID (sys_id) attribute to identify services on the source and target instances and implements service definition import according to these rules:

- If a service Sys ID does not exist on the target instance, the system creates a new service on the target instance.
- If a service Sys ID and name already exist on the target instance, the system updates this service by overriding the existing service definitions with the imported definitions.
- If a service Sys ID exists on the target instance, but is associated with a different service name, the system updates the service with this Sys ID. The system overrides service definitions, including the service name.
- If the same name is used for different services on the source and target instances, the system does not import service definitions.
- You cannot export manual services. Likewise, you cannot export manually added entry points. When importing definitions of a service containing manual entry points, the system does not make any changes to the entry points.

When transferring service definitions between instances, imported application services and service groups are assigned to the user's domain.

By default, the system assigns **Non-operational** status to application services whose definitions you imported.

A service definition transfer deploys the standard import set feature available on the Now Platform. When you export service definitions on the source instance, the system creates a JSON file containing exported data. When you copy service definitions into the target instance, you import the data from this JSON file, which becomes an import set.
Export service definitions
Create an import set containing definitions of services. You use this import set to import service definitions into another instance.

- Decide which service definitions you want to export.
- Verify that the source and the target instances are on the same ServiceNow release.
- Open the source instance from which you want to copy service definitions.
- If necessary, fine-tune the import set creation by modifying relevant properties under Service Mapping > Administration > Properties. For more information, see Components installed with Service Mapping.

Role required: sm_admin

1. If the instance uses domain separation and you have access to the global domain, select the domain to which the application service belongs from the domain picker.
3. Select the application services that you want to export.
4. Click Actions on selected rows at the bottom of the page and select Export Services.
   The import set appears in the Downloads folder. The file name follows this convention: export_service_<datetime>.json.
5. If necessary, copy the import set to a different location on the computer.

Import service definitions from one instance to another
Import definitions of application services from the source instance.

- Verify that the source and the target instances are on the same ServiceNow release.
- Open the target instance into which you want to import service definitions.

Role required: import_admin

1. If the instance uses domain separation and you have access to the global domain, select the domain to which the application service belongs from the domain picker.
3. In the Import Service Definitions window, click Choose File, navigate to the JSON file that contains service definitions, and click OK.
4. Optional: Click Show Preview and check the definitions in the chosen JSON file.
5. Click Import.
6. Click Yes to confirm that you are ready to import service definitions.

Attention: During import, the system irreversibly overrides the current service definitions.

The Progress window displays the success message.
Service Mapping automatically starts the discovery of services containing imported entry points.
7. Review imported or updated services after Service Mapping finishes the mapping process. If the transfer result is not as you expected, perform basic troubleshooting as described in the following table.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
</table>
| Changes you have recently made to an application service disappear. | Definitions of existing application services are always overridden during transfer. | 1. Make necessary changes to the definitions of this application service again.  
2. Run the top-down discovery of this application service by clicking Run Discovery in the service map. |
| An imported service group does not appear.         | The import of an application service assigned to this group failed and the system did not create the group because it appears to be empty. | 1. Resolve the issue that prevented the system from importing the application service belonging to this group.  
2. Import the import set again.  
3. Check that the service group appears. |
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
</table>
| An imported service group appears to have wrong application services assigned to it. | This is a different service group with the same name, not the service group you imported. The system failed to import the service group, because the source instance already has a service group with the same name or with the same name and Sys ID. | 1. Check the import log.                                                                                                                                  
2. If an error states that a record creation failed while running the Check Name Uniqueness business rule, perform the following steps:
   a. Rename the service group on the source instance.
   b. Re-export the application services assigned to this group.
   c. Import the latest import set on the target instance.
3. If an error states that a record creation failed because the Sys ID already exists, perform the following steps:
   a. Recreate the service group on the source instance, ensuring you provide a different group name and Sys ID to it.
   b. Re-export the application services assigned to this group.
   c. Import the latest import set on the target instance.
4. Verify that the service group appears correctly on the source instance. |

**Application service analysis and maintenance using maps**

Service Mapping creates maps to help you see the architecture and organization of application services. These maps are useful for planning change or migration, as well as analyzing the continuity and availability of services.

The maps are useful in the following cases:

**Planning the migration of an entire application service or its segments**

Use the map to see which configuration items (CIs) need relocating or which CIs become redundant.

**Planning the upgrade or replacement of a CI**
Check the map to see what other CIs are affected when the CI is non-operational during maintenance. You can assess and plan the downtime or make provisions to avoid the downtime.

Because the same CI may be used for multiple application services, check the CI using the Dependency Views application. With this application, you can see all application services that a CI belongs to.

**Analyzing a application service for high availability and continuity**

Use the map to identify CIs crucial for the service performance. Decide if you want to fortify these CIs by creating clusters.

**Troubleshooting application services**

Use the map to understand the impact of an issue and determine which CI is causing the problem.

**Application service maps**

Maps offer you a visualization of data on configuration items (CIs) comprising application services, and relations and connections between these CIs.

A new map is created every time you manually create an application service or map an application service using Service Mapping. The system updates the map every time you add CIs to the application service or when Service Mapping runs discovery for CIs belonging to this application service.

**Note:**

The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

Information and actions available on a map depend on your role:

- As a Service user (app_service_user) or Service Mapping user (sm_user), you can view the map in the View mode.
- As a Service administrator (app_service_admin) or Service Mapping administrator (sm_admin), you can view application services in View mode and modify services in Edit mode.
- As a Service Mapping administrator (sm_admin), you can view discovery messages and errors generated by Service Mapping and modify services in the Edit mode. You can also reload the map to ensure the map reflects the latest information.
- As a Service owner (sm_app_owner) role, you can review, provide feedback, and approve services discovered using Service Mapping in the View mode.

Also, actions and information related to automatic discovery using Service Mapping are not available for CIs that Service Mapping cannot discover. For example, you do not see the tab for Discovery Messages for a manually created application service that Service Mapping does not discover.

**Map window**

Aside from the map itself, the map window also displays the Properties pane and tabs with additional information.
You can navigate to a different application service map directly from this window by selecting it from the list:
By default, the entire map is shown in the center of the visible map area of the window. You can zoom in and out as well as position the map depending on which segment of the map you must see using these controls:

You can also click anywhere in the map area and drag the required segment of the map into the visible area.
You can view changes made to an application service as a whole and to individual CIs belonging to a service by choosing a time range. For more information, see View the change history of application services.

**Map elements and their appearance**

Every map consists of icons representing CIs and arrows that represent the connections between them.

Some map elements can contain other elements such as in the case of application clusters. On the map, application clusters appear as a stack of CIs with the plus (+) symbol next to it. The cluster label shows the number of CIs in this cluster with the multiplication sign.

When the CI has inclusions, the CI icon has the plus (+) symbol next to it and shows the contained application when you expand it.
An OS cluster appears as a CI with the plus (+) symbol and the number of CIs in this cluster.

**CI attributes and CI-related messages**

When a map is loaded and no elements are selected, the Properties pane shows the details of this application service.

A selected device, application, or connector appears in blue and is highlighted. Information about the selected map element is displayed in the Properties pane on the right of the map. The Properties pane contains the links to the Detailed Properties page.
The attributes for the server hosting applications and applications themselves are shown separately inside the Properties pane.
If there is any information related to the selected CI or connection, it is highlighted on the tab at the bottom of the window.
When you select information on a tab at the bottom of the map window, the related CI displays in yellow.
Map indicators

You can show additional information for a CI or application service itself by displaying its related records such as alerts, outages, incidents, and problems. If you enable this feature, record indicators appear next to relevant CIs and next to the application service name. Records themselves are displayed on the tabs at the bottom of the map window.
Map views

Customize a map view to exclude irrelevant information and increase clarity, as described in Modify view for an application service map.

To open this application service in Event Management and view alerts on CIs that are part of the application service, click **Monitor Service** at the top of the window.

View CI attributes in an application service map

An application service map displays attributes for each configuration item (CI) that is part of the application service, as well as for the application service itself. The attributes come from the CMDB.

Role required: sm_admin or sm_user

You can view the following information for each CI:

**Name label**
The CI name. This attribute is either preconfigured on the CI or configured during CI installation.

**Basic attributes**
A summary of the most important CI attributes.

**Detailed attributes**
A complete list of all attributes collected for the CI.

Each CI type (CI class) has different attributes. For example, the Linux Server type has different attributes than the SQL Instance type.

If Service Mapping is deployed, the way CIs appear on the map depends on the **view you select for the map**. Attributes available for viewing also depend on the Service Mapping setup. For more information see description of **components installed with Service Mapping**.

The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

1. Navigate to **Configuration > Application Services > Application Services**.
2. Alternatively, if Service Mapping is deployed, navigate to **Service Mapping > Services > Application Services**.
3. Click **View Map** next to the relevant application service.
4. To see the full name of a CI whose name has been shortened on the map, point to the CI. A tooltip displays the full CI name.
5. Click a CI to see its details in the **Properties** pane. The attributes of applications and the servers that host them appear separately.
6. To view more detailed attributes for the CI, click **Detailed properties** at the bottom of the Properties pane.

7. Optional: (If Service Mapping or Discovery are enabled, and if tracking changes to configuration files is enabled for a CI) To view configuration files associated with a CI:
   - Review the list of files under **Tracked Files** in the Properties pane. Click the file name to open the actual file.
   - Click the **Affected CIs** tab and view the list of configuration files. Click the file name to open the actual file.

**View CI connection attributes in an application service map**

You can view attributes of connections between configuration items (CIs).

Role required: sm_admin or sm_user
You can view the following information for each connection:

- The source and target CIs of this connection displayed in the Context (right-click) menu.
- A complete list of all attributes collected about the connection in the Properties pane.

By default, Service Mapping merges connection lines for the same CI to declutter an application service map. It helps to make the map more readable.

Each connection type has different attributes. For a merged connection line, all underlying connections are listed.

The way an application service map displays connection attributes depends on the following factors:

- **Views you configure for CIs** define which connection attributes appear in the Properties pane of the map.
- Connection lines on the map depend on the **view you select for the map**.
- Attributes available for viewing also depend on the Service Mapping setup, for more information see description of **components installed with Service Mapping**.
The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

1. Navigate to **Configuration > Application Services > Application Services**.
2. Alternatively, if Service Mapping is deployed, navigate to **Service Mapping > Services > Application Services**.
3. Click **View map** next to the application service including the connections you want to view.
4. View connection attributes as follows:

<table>
<thead>
<tr>
<th>To view</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>The source and target CIs of a regular connection</td>
<td>Right-click the connection.</td>
</tr>
</tbody>
</table>

![Diagram of connection attributes](image)
<table>
<thead>
<tr>
<th>To view</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Available for application services discovered with Service Mapping) The source and target CIs for a merged connection</td>
<td>Right-click the merged connection line with a number and select the relevant connection.</td>
</tr>
</tbody>
</table>

- Right-click the merged connection line with a number and select the relevant connection.

Or

- If a segment of a connection line is shared by more than one merged connections, there is no indication that it is a merged line.

Right-click the connection line coming out or going into a CI, and select the relevant connection.
<table>
<thead>
<tr>
<th>To view</th>
<th>Do this</th>
</tr>
</thead>
</table>
| The source and target CIs for a connection if the spanning tree view is enabled. The spanning tree view simplifies an application service map concealing most of the connection lines. | 1. Click the CI whose connections you want to view. All concealed connections for this CI appear on the map.  
2. Right-click the relevant connection line. |
| Detailed attributes of a regular connection in the Properties pane      | Click the connection line.                                                                                                               |
To view

(Available for application services discovered with Service Mapping)
Detailed attributes of a merged connection in the Properties pane.

Do this

Click the merged connection line indicated by a number.
All underlying connections are displayed separately in the Properties pane.
<table>
<thead>
<tr>
<th>To view</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detailed attributes of a connection segment shared by more than one merged connection in the <strong>Properties</strong> pane.</td>
<td>Right-click the connection line coming out or going into a CI, select the relevant connection, and then choose <strong>Select edge</strong>.</td>
</tr>
</tbody>
</table>

The detailed attributes are displayed in the **Properties** pane.

**Note:** Shared segments do not have an indication that it is a merged line.
<table>
<thead>
<tr>
<th>To view</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Available for application services discovered with Service Mapping) Detailed properties of a connection in the <strong>Properties</strong> pane, when the spanning tree view is enabled. The spanning tree view simplifies an application service map concealing most of the connection lines.</td>
<td>1. Click the CI whose connections you want to see. All concealed connections for this CI appear on the map. 2. Click the required connection line.</td>
</tr>
</tbody>
</table>
To view Traffic-based connections for a CI.

1. Right-click the CI whose traffic-based connections you want to check.
2. Select Show traffic based connections.
   The Traffic Based Connections List opens displaying the information in Traffic Based Connections List table.
3. Click Close when finished viewing this list.

Note: You can see traffic-based connections for a CI, even if you disabled the traffic-based discovery for an application service the CI is part of.

Manually added connection.
You can either add a connection or a CI manually. In both cases, the connection you manually add an entry point to the next tier of the application flow.

Click the manually added connection.
The Properties pane shows Manual Endpoint under Endpoint Type.

Traffic Based Connections List

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>The IP address of the application connected to the selected CI.</td>
</tr>
<tr>
<td>Port</td>
<td>The port on the selected CI that is used to communicate to the other application.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>The ID of the process in the selected CI.</td>
</tr>
</tbody>
</table>
| Already on map         | - **Yes** — if this connection shows on the map.  
- **No** — if this connection is not part of the application service and not on the map.                                                   |
| System decision        | The setting defines if Service Mapping keeps the discovered traffic-based connection or removes it. The value comes from the algorithm that Service Mapping uses. |
| User decision          | (Optional) The setting overrides the System decision setting, which defines if Service Mapping keeps the discovered traffic-based connection or removes it. For example, if the System decision setting for a connection is Keep, and you want to remove this connection, select **Remove**. |

### View the change history of application services

You can view changes made to an application service as a whole and to the individual configuration items (CIs) comprising the service. Change history is useful for maintenance, planning, or troubleshooting procedures.

Role required: admin, sm_admin, sm_user, app_service_admin, or app_service_user

Details about changes to an application service and to its CIs is stored in the CMDB. Typically, these changes reflect adding or removing CIs from an application service, upgrading or updating CIs, or modifying CI configuration files. The system gathers this data by querying CMDB tables and then creating the change history view. In deployments where Service Mapping is activated, the type of change information Service Mapping queries depends on discovery patterns that Service Mapping uses to discover CIs.

Changes to configuration files are associated with CIs to which these files belong. Maps show configuration file changes as changes to related CIs.

While you can see change records for a specific CI in the context of application services, you can also see detailed history of a specific CI separate from its application service as described in [History Timeline](#).

If the Now Platform is configured to validate changes, all changes are evaluated and rendered as valid or not. If a change is valid, its change record on the application service map is marked as approved. For more information about configuring the platform for change validation, see [Managing proposed changes](#).

Changes to the application service appear on the history timeline.
The type of change mark depends on the nature of changes that it represents:

- **Light gray balloon**
  Unapproved change that does not influence the application service behavior. For example, a change in a network path or adding a node to a cluster.

- **Dark gray balloon**
  Unapproved change that changes the application service behavior.

- **Green balloon**
  An approved change in deployments where the Now Platform is configured to validate changes.

- **Double balloon**
  Multiple separate changes that happened a short time from each other.

You can mark times on the history scale by creating baselines to quickly return to the marked view.

The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

1. Navigate to **Configuration > Application Services > Application Services**.
2. Alternatively, if Service Mapping is deployed, navigate to **Service Mapping > Services > Application Services**.
3. Click **View map** for the application service for which you want to see change history. Actions or map elements which are relevant only for application services discovered by Service Mapping, do not appear on maps for manually created application services.
4. On the history timeline, set the time range of changes that you want to view.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>To set the time range of the history timeline</td>
<td>Click the hour, day, week, or month icons.</td>
</tr>
<tr>
<td>Option</td>
<td>Action</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>To increase or decrease the time range</strong></td>
<td>Click the zoom in and zoom out icons.</td>
</tr>
<tr>
<td><img src="image" alt="Zoom in and zoom out icons" /></td>
<td><img src="image" alt="Zoom in and zoom out icons" /></td>
</tr>
<tr>
<td><strong>To change the upper limit on your history range</strong></td>
<td>Click the history scale.</td>
</tr>
<tr>
<td><img src="image" alt="History scale" /></td>
<td><img src="image" alt="History scale" /></td>
</tr>
</tbody>
</table>

The time that serves as the upper limit appears above the history timeline.

**Note:**
You cannot set the lower limit on your history range to a time before this application service was created. This time is marked with the IT Service Created event on the history timeline.

The map shows the history view of the application service for the time you selected.

**Note:** The Change tab shows all change records, even the ones which are filtered out of the history view.

5. To mark a time on the time scale, set a baseline:
   a) Click the Compare icon.
   b) Navigate to the time you want to mark as a baseline on the history scale.
   c) Click Set baseline.
d) Enter the name of the baseline and click **OK**.
The new baseline appears as a button above the history scale and as a blue flag on the history scale.

6. View the change history:
Option | Action
--- | ---
To see the CI responsible for a change record | Select a change record on the Changes tab. The related CI is marked yellow in the map.
<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
</table>
| **To see only change records related to a CI** | Select the required CI or the connection on the map.  

The Changes tab displays only change records related to the selected CI or connection. |

| **To see the configuration file at the selected moment in the past** | 1. Set the time on the history scale.  
2. In the Properties pane, scroll to Tracked Configuration Files, and click the file name.  

The new tab opens displaying the content of the tracked configuration file at the selected time. |
To see network at the selected moment in the past

1. Set the time on the history scale.
2. Right-click the connection and select Show network path.

The new tab opens displaying the network or storage path map for the time you selected.

Note: You cannot view the network path for connections marked as boundaries to this application service.

7. To exit the history view and see the current status of the application service, click the current icon.

**Compare two versions of an application service**

You can see a summary of application service changes at a glance by comparing two versions of an application service. This feature is useful for checking the application service status before and after a certain change or problem.

Role required: admin, app_service_admin, app_service_user, sm_admin, or sm_user

Specify two points in time for which to compare the two versions of an application service. You can use the change indicators on the timeline to specify one point in time that is before and another that is after a change for which to see the details. For example, if you know that the application service started to fail at a certain time, you can compare two versions of the application service, one before and one after the problem started. This comparison lets you see the summary of changes that possibly led to the problems.

Service Mapping, if deployed, tracks and shows all changes to a CI including configuration files associated with a CI. When you compare two versions of an application service, you can see changes made to configuration files as changes to CIs. You can also compare two versions of a configuration file to see the actual changes in the files, during the time range specified for the comparison.

The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

1. Navigate to Configuration > Application Services > Application Services.
2. Alternatively, if Service Mapping is deployed, navigate to Service Mapping > Services > Application Services.
3. Click View Map next to an application service.
4. On the history timeline, set the time range of changes that you want to view.

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>To set the time range of the history timeline</td>
<td>Click the hour, day, week, or month icons.</td>
</tr>
<tr>
<td>Option</td>
<td>Action</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>To increase or decrease the time range</td>
<td>Click the zoom in and zoom out icons.</td>
</tr>
<tr>
<td>To change the upper limit on your history range</td>
<td>Click the history scale.</td>
</tr>
</tbody>
</table>

The time that serves as the upper limit appears above the history timeline.

**Note:**
You cannot set the lower limit on your history range to a time before this application service was created. This time is marked with the IT Service Created event on the history timeline.

The map shows the history view of the application service for the time you selected.

**Note:** The Change tab shows all change records, even the ones which are filtered out of the history view.

5. Click the Compare icon.

6. Set **Compare point 1** and **Compare point 2** as the two points in time for the comparison.

You can drag the pointers on the history scale to set corresponding time points.
If the history scale does not include the time set for comparison, then its corresponding pointer appears next to the compare point in yellow:

![Image of history scale with yellow pointer]

**Note:** If there are no changes to the service during the time interval specified by **Compare point 1** and **Compare point 2**, then no change details are displayed.

7. **Click Compare.**  
   The comparison view opens in a separate tab.
8. **Select a marked CI to see the relevant change record on the Changes tab.**
9. Optional: If Service Mapping is deployed, you can compare two versions of a configuration file that appears on the map as Updated:
   a) Select the CI that is associated with the updated configuration file.
   b) In the Properties pane, click the link to the updated file.
The **Tracked Configuration Files Version Compare** tab opens showing two versions of the configuration file side by side.

c) **Review actual changes.**

Highlight colors indicate the type of change:
- Purple — Updated line
- Pink — New line
- Gray — Deleted line

d) **Navigate between the changes using the arrows in the upper right corner.**

e) **Close the Tracked Configuration Files Version Compare** tab when finished.

10. Close the comparison view when finished.
Check CI dependencies

You can see if a particular configuration item (CI) is part of other application services and check if it depends on other CIs.

Role required: admin, app_service_admin, app_service_user

While Service Mapping shows position of a CI in a particular application service, your organization may also use the same CI in other application services. If you are planning to perform changes to this CI, make sure that the changes do not affect other application services.

The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

1. In the map, right-click the relevant CI and select **Open dependency views**.
   The Dependency Views page opens in the new tab showing the selected CI in the center.

2. Click **Details**.

3. Click the **Related Services** tab.
   The list of links to application services containing this CI appear on the **Related Services** tab.

4. To view the map of another business service containing this CI, click the relevant link.

View network or storage path

You can drill down to see objects behind a connection on a map for troubleshooting service performance or maintaining your network. This feature is not available for manually created services or for services on instances using Edge Encryption.

Role required: admin, sm_admin or sm_user

Displaying a network or storage path allows you to see routers, switches, and host ports which are part of the connection. The following connection types exist:

**Network path**

The infrastructure route between two CIs of the host type.

**Storage path**

The infrastructure route between a CI of the host type and a CI of the storage type.

Service Mapping discovers and displays network devices located on your organization local network, not on WAN or the public cloud. When Service Mapping encounters a connection between two applications, it uses traceroute on the source host to figure out the path to the target host. Then Service Mapping use SNMP credentials to discover the devices it found on the path.

Service Mapping can show devices for storage paths using the following protocols:

- Fiber Channel: Storage Area Network (SAN)
- IP Ethernet: Internet Small Computer Systems Interface (iSCSI) or Network Attached Storage (NAS)

**Note:** For storage paths using Fiber Channel, Service Mapping does not display Fiber Channel switches.
Between a host CI and a storage CI there may be two separate storage paths using different protocols: Fiber Channel and IP Ethernet. If so, the application service map displays one connection line for both protocols.

You can see the change history for network and storage paths for application services. Checking path history for manual or technical services is not supported.

1. Navigate to **Service Mapping > Services > Application Services** and click **View map** next to the relevant application service.
2. Alternatively, navigate to **Event Management > Dashboard** and double-click an application service.
3. On the map, right-click the connection for the network or storage path you want to see.
4. Select **Show network path** or **Show storage path**.
5. If necessary, select the type of protocol and click **Choose** in the Choose Protocol Type dialog box.
   The view showing the network or storage path opens in a new tab.

If Service Mapping cannot discover all objects in a connection, the connector line on the network or storage path appears dotted as shown above.

6. For storage paths, use the Storage Mapping tab at the bottom of the screen to view correlation between file systems on the host and the storage volume or the shared folder on the storage device.
7. To view the change history for network paths, perform steps in **View the change history of application services**.
8. If you access the network or storage path map from Event Management, you can see the list of alerts related to this path appear at the bottom of the screen.
Convert application service maps into PDFs

You may want to create a PDF for a map to share information about the service content with other people. PDFs are especially useful during the review and approve process for application services discovered by Service Mapping.

Role required: app_service_admin, app_service_user

The PDF of an application service map shows the name of the application service and the exact time when this PDF was created.

1. Navigate to Configuration > Application Services > Application Services.
2. Alternatively, if Service Mapping is deployed, navigate to Service Mapping > Services > Application Services.
3. Click View map next to the relevant application service.
4. Click the Additional actions menu.

5. Click the button next to the Export to PDF option.
6. When the PDF is ready, click the **Download PDF** button in the Export Map into PDF Format dialog box.
   The PDF appears in the downloads folder on your computer.

**Modify view for an application service map**

Customize a map view to exclude irrelevant information and increase clarity.

Role required: app_service_admin, app_service_user, sm_admin, or sm_user

By default, service maps are concise and contain every detail about all CIs that you added manually or Service Mapping mapped. While it provides a complete picture, using such a map might be difficult.

Customizing maps allows you to:
- make it easier to navigate the map.
- hide irrelevant information.
- show essential information.
Attributes you show or hide when you customize maps are not removed permanently and can be displayed or hidden again as you need. The system saves the changes to the map view and displays the last map view when you reopen an application service.

The following user interface screens refer to application services as business services: Business Service Group Responsibilities, Approve, Service Mapping Properties, and Service Map Planner.

1. Navigate to Configuration > Application Services > Application Services.
2. Alternatively, if Service Mapping is deployed, navigate to Service Mapping > Services > Application Services.
3. Click View map next to the relevant application service.
4. Click the Additional actions menu.

5. Select the relevant option:
<table>
<thead>
<tr>
<th>Option</th>
<th>When Enabled</th>
<th>When Disabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display in Host View</td>
<td>The service map shows hosts.</td>
<td>The service map shows CIs.</td>
</tr>
<tr>
<td></td>
<td><img src="#" alt="ServiceMap Diagram" /></td>
<td><img src="#" alt="ServiceMap Diagram" /></td>
</tr>
<tr>
<td></td>
<td>Click the plus icons to see CIs running on the hosts.</td>
<td>The Properties pane shows information about both servers who act as hosts.</td>
</tr>
<tr>
<td></td>
<td>The Properties pane shows only information about servers who act as hosts.</td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>When Enabled</td>
<td>When Disabled</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Group CIs on map</strong></td>
<td>The application service map shows groups of CIs. The map is more compact.</td>
<td>The application service map shows all CIs separately. The map is larger.</td>
</tr>
<tr>
<td></td>
<td>Simplify maps by grouping CIs belonging to the same type and hosted on servers sharing prefix and domain name.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The map still reflects the application flow correctly, because grouped CIs belong to the same tier.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>This option is relevant for application services consisting of more than 10 CIs.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The application service map shows groups of CIs. The map is more compact.</td>
<td></td>
</tr>
<tr>
<td>Remove Topology Cycles</td>
<td>The application service map hides topology cycles.</td>
<td>The application service map shows topology cycles.</td>
</tr>
<tr>
<td></td>
<td>This option is available only for users with these roles: Service Mapping administrator (sm_admin) and Service Mapping user (sm_user).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hide connections returning to the same configuration item (CI) that originated them to reduce map cluttering.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Usually these returning connections are not important for analysis or troubleshooting of application services.</td>
<td></td>
</tr>
<tr>
<td>Show Traffic Based CIs</td>
<td>The application service map displays CIs and hosts that Service Mapping discovered using both patterns and traffic-based discovery.</td>
<td>The application service map shows only CIs and hosts discovered using patterns.</td>
</tr>
<tr>
<td></td>
<td>While detecting CIs inbound and outbound traffic creates an inclusive map, it may also result in mapping many redundant CIs that do not influence the application service operation.</td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>When Enabled</td>
<td>When Disabled</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Merge error nodes</strong></td>
<td>This option is available only for users with the Service Mapping administrator (sm_admin) role. Merge CIs with errors caused by the same source CI to declutter an application service map. A discovery or operational error in a CI is often propagated to all its connections down the hierarchy crowding your view. You can remove identical errors caused by the same CI by consolidating them in one object on the map. For example, if one CI causes an error in six CIs that are connected to it, you can show only one CI with this error on the map. The connector line shows the number of actual connections. The application service map merges all CIs with identical errors into one. The number on the connector line indicates the number of CIs with this error. The application service map shows all CIs with identical errors separately.</td>
<td></td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Option</th>
<th>When Enabled</th>
<th>When Disabled</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Map indicators</strong></td>
<td>For each record type that is set to display, the corresponding indicator and the tab display. For example, the <strong>Incidents</strong> tab appears underneath the map and the <strong>Incidents</strong> indicator shows next to CIs that have incidents associated with them.</td>
<td>The application service map shows CIs without related indicators.</td>
</tr>
<tr>
<td></td>
<td>Display additional information for a CI or application service itself by displaying related records such as alerts, outages, incidents, and problems.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The base system provides map indicators for the following record types:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Affected CIs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Open alerts</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Current, planned, or recent change requests</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Open incidents</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Current, planned, or recent outages</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Open problems</td>
<td></td>
</tr>
<tr>
<td></td>
<td>For more information, see <a href="https://www.servicenow.com/servicenow">Event Management Map Indicators (Video)</a>.</td>
<td></td>
</tr>
</tbody>
</table>

Map indicators
This option is available only for users with these roles: Service user ([app_service_user]) and Service Mapping user ([sm_user]).

Display additional information for a CI or application service itself by displaying related records such as alerts, outages, incidents, and problems.

The base system provides map indicators for the following record types:

- Affected CIs
- Open alerts
- Current, planned, or recent change requests
- Open incidents
- Current, planned, or recent outages
- Open problems

For more information, see [Event Management Map Indicators (Video)](https://www.servicenow.com/servicenow).
<table>
<thead>
<tr>
<th>Option</th>
<th>When Enabled</th>
<th>When Disabled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>If a record is associated with the application service itself, the indicator shows next to the application service name.</td>
<td>No map indicators appear next to the application service name.</td>
</tr>
<tr>
<td>Option</td>
<td>When Enabled</td>
<td>When Disabled</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Spanning tree view</strong></td>
<td>The map displays an application service as a tree.</td>
<td>The map reflects the actual structure of an application service.</td>
</tr>
<tr>
<td></td>
<td>To view actual connections, point to a CI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>For more information on viewing connections, see <a href="#">View CI</a>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>connection attributes in an application service map.</td>
<td></td>
</tr>
<tr>
<td><strong>Include ignored errors</strong></td>
<td>The map shows all errors: both active and ignored.</td>
<td>The map does not show ignored errors.</td>
</tr>
<tr>
<td></td>
<td>You can display errors which were previously hidden from the map.</td>
<td></td>
</tr>
</tbody>
</table>

**Keyboard shortcuts for operating Service Mapping in accessibility mode**

You can access all application service maps and service forms using a keyboard in the accessibility mode.
When the whole map is selected, the information message at the top of the window offers guidelines on how to navigate inside the map.

<table>
<thead>
<tr>
<th>Action</th>
<th>Keyboard shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access right-click menus.</td>
<td>Space</td>
</tr>
<tr>
<td>Expand clusters.</td>
<td>Enter</td>
</tr>
<tr>
<td>Move into the map pane from the Properties</td>
<td>Escape</td>
</tr>
<tr>
<td>pane.</td>
<td></td>
</tr>
<tr>
<td>In the More Options menu, toggle options on and off.</td>
<td>Space</td>
</tr>
<tr>
<td>Select CIs and connections inside the map.</td>
<td>Arrow</td>
</tr>
<tr>
<td>In the timeline, you can edit the time inside the current time selection.</td>
<td>Number (1-9)</td>
</tr>
<tr>
<td>Move through UI elements in the application service map window.</td>
<td>Tab</td>
</tr>
</tbody>
</table>

**Advanced Service Mapping configuration**

Fine-tune Service Mapping collaboration with other components and modules as well as customize data display in application service maps.

In addition to the basic setup described in [Service Mapping setup](#), you can perform optional configurations to meet the needs of your organization.

**Enable traffic-based discovery for CI types or specific CIs**

Service Mapping can discover and map CIs by detecting the inbound and outbound traffic that the CIs generate. Create a traffic-based discovery rule to determine which configuration items are available for traffic-based mapping.

If your ServiceNow instance uses domain separation and you have access to the global domain, select the domain to which the application service belongs from the domain picker. The selected domain must be a domain without any child domains.

Role required: sm_admin

- Ensure that the traffic-based discovery is enabled at the Service Mapping product level: navigate to **Service Mapping > Administration > Properties** and verify that the **Traffic based discovery** check box is selected.
- Enable traffic-based discovery for a specific application service as described in **Map a single application service**.

You may choose to use traffic-based discovery and mapping in addition to the pattern-based mapping.

You can create rules to configure Service Mapping to use traffic-based discovery for certain CIs. You can configure traffic-based discovery rules to monitor specific CIs or types of CIs.

Rules for specific CIs take precedence over rules for CI types. For example, if you do not want to use traffic-based discovery on any Tomcat servers, you can define a **CI type** rule disabling the traffic-based discovery on the Tomcat table. At the same time, you can create a discovery rule enabling the traffic-based discovery for a specific Tomcat server. In that case, Service Mapping uses the traffic-based discovery only for this specific Tomcat server out of all Tomcat servers.
If you used both pattern-based and traffic-based mapping for an application service, its map displays CIs and hosts that Service Mapping discovered using both methods. While detecting CI inbound and outbound traffic creates a very inclusive map, it may also result in mapping many redundant CIs that do not influence the application service operation.

If your instance uses domain separation, you can create traffic-based rules for specific domains. Rules in the base system are assigned to the global domain and apply to all domains of all levels.

When you create a rule for a specific domain, the new rule is used only for this domain and does not exist in any other domains. If you customize an existing rule in the global domain and assign it to a specific domain, you create a copy of the global rule, which is still used in all other domains except the domain that has the customized version of this rule. Likewise, if you customize a rule in the global domain, the change affects all domains except the one that uses a customized copy of this rule.

1. Navigate to **Service Mapping > Administration > Traffic Based Discovery**.
2. Click **New**.
3. Define the rule parameters as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action</strong></td>
<td>Select <strong>Enable</strong> to add traffic-based connections to the specific CI or CI Type.</td>
</tr>
<tr>
<td><strong>Rule Scope</strong></td>
<td>Select <strong>Specific CI</strong> to detect traffic for one configuration item, or select <strong>CI Type</strong> to detect traffic for all configuration items in one of the CI-based table.</td>
</tr>
<tr>
<td><strong>CI/CI Type</strong></td>
<td>Select a specific CI, or select the table that contains the CIs for which you want to detect traffic.</td>
</tr>
<tr>
<td><strong>Domain</strong></td>
<td>Select the domain to which the rule belongs.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.
5. To fine-tune traffic-based discovery, define advanced parameters as follows:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| **sa.traffic_based_discovery.conn_aging_time** | Time period in hours for a Traffic Based Connection to remain active since last discovered.  
  - **Type**: integer  
  - **Default value**: 72  
  - **Other possible values**: any number higher than 24  
  - **Location**: Service Mapping > Administration > Properties. |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sa.traffic_based_discovery.ignored_ports</td>
<td>Ports to ignore when found by traffic-based discovery. This property is available in the System Property (sys_properties) table. Change this property to define ports that Service Mapping ignores while performing traffic-based discovery. It makes discovery more efficient since resources are not wasted on discovering irrelevant connections.</td>
</tr>
</tbody>
</table>
| | · **Type**: string  
| | · **Default value**: 445, 139, 111, 2049, 860, 3260, 135, 53  
| | · **Other possible values**: any relevant port numbers  
| | · **Location**: System Property (sys_properties) table |
| sa.traffic_based_discovery.max_connections | Maximum number of traffic-based connections from a single CI. This property is available in the System Property (sys_properties) table. This property helps to keep the map size reasonable by limiting the number of possible CI connections. |
| | · **Type**: integer  
| | · **Default value**: 30  
| | · **Other possible values**: any number higher than 1  
| | · **Location**: System Property (sys_properties) table |

### Configuration file tracking

The horizontal discovery process can find configuration files that belong to certain applications and add those configuration files to the CMDB. You can track the changes to these files by comparing them to previous versions.

**Warning**: Configuration files contain sensitive system information. To prevent unauthorized access, ensure that access control lists (ACL) are placed on the Tracked Configuration file table (cmdb_ci_config_file_tracked). Only allow authorized users to view this table or uncheck the **Save Content** setting.

### Components for configuration file tracking

#### CI type

All applications and hosts in your organization must have a corresponding configuration item (CI) type, which is necessary for discovering and processing applications and hosts correctly. In a base system, many CI types have configuration file paths defined for them. You can add
new or modify existing definitions for tracking configuration files. See Modify tracking changes in configuration files for instructions.

**Patterns**

Configuration file tracking is available for patterns that discover applications. On the pattern, you can create tracked file definitions that specify the CI type to which the application CI belongs and the path of the configuration file. Specify as many tracked file definitions as needed. You can also specify whether you want to save the contents of configuration files so you can view and compare the contents of different versions.

**Note:**

Configuration file tracking is not available for discoveries performed by traditional probes and sensors. The classifier that triggers the pattern must specify the Horizontal Pattern probe, which in turn, must specify the pattern. If you upgrade your instance to the current version, not all classifiers are configured to use patterns for discovery by default.

**CMDB**

All configuration files are saved as a CI in the Tracked Configuration file (cmdb_ci_config_file_tracked) table. If you enable the content to be saved, these CI records provide the contents of the configuration files, including previous versions. From the configuration file CI record, you can compare different versions.

**Dependency maps and application service maps**

Both dependency maps and application service maps display tracked configuration files. The relationship between a configuration file and its host is a contains relationship. The application contains the configuration file.

For example, this IIS web server contains three tracked configuration files:

![Dependency map diagram]

Sometimes you organize CI types as a main CI type and its related CI types. On an application service map, Service Mapping shows changes to configuration files of related CIs for the main CIs in inclusions. In inclusions, the system treats applications hosted on a server as independent objects. For example, the Tomcat WAR CI appears separate from its host, the Tomcat CIs. In this case, Service Mapping shows changes to configuration files of Tomcat WAR when you select
Tomcat. In addition, Service Mapping displays changes to configuration files of the hardware server hosting inclusions. In this example, it is a Linux server:

**Deletion strategy**

You can specify what you want to do with tracked configuration file CI records when discovery can no longer find them. You can keep the configuration file CI record, automatically delete it, delete only the CI relationships to it, or mark it absent.

**Discovery patterns that support configuration file tracking by default**

These patterns provide tracked file definitions by default:

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Pattern</th>
<th>CI Type</th>
<th>File path of tracked file</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Apache Server</strong></td>
<td>Apache On Unix Pattern</td>
<td>Apache Web Server</td>
<td>$config_file</td>
</tr>
<tr>
<td></td>
<td>Apache On Windows Pattern</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MySQL Server</strong></td>
<td>MySQL server On Windows and Linux Pattern</td>
<td>MySQL Instance</td>
<td>$config_file</td>
</tr>
<tr>
<td><strong>Microsoft IIS Server</strong></td>
<td>IIS</td>
<td>Microsoft iis Web Server (cmdb_ci_microsoft_iis_web_server)</td>
<td>EVAL(javascript: var rtrn = '';var winDir = CTX.getCommandManager().shellCommand(&quot;echo %WinDir%&quot;);rtrn = winDir.trim() + '\System32\Inetsrv\Config*.config&quot;);)</td>
</tr>
<tr>
<td></td>
<td>IIS Virtual Directory</td>
<td></td>
<td>$install_directory + &quot;*.config&quot;</td>
</tr>
</tbody>
</table>
## Table: Configuration File Tracking

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Pattern</th>
<th>CI Type</th>
<th>File path of tracked file</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Matrix Business Works</td>
<td>Active Matrix Business Works</td>
<td>cmdb_ci_appl_tibco_matrix</td>
<td>$config_file</td>
</tr>
<tr>
<td>Enterprise Message Service</td>
<td>Enterprise Message Service</td>
<td>cmdb_ci_appl_tibco_message</td>
<td>$config_file</td>
</tr>
<tr>
<td>Oracle</td>
<td>Oracle DB on Windows Pattern</td>
<td>cmdb_ci_db_ora_instance</td>
<td>$install_directory + &quot;/network\admin*.ora&quot;</td>
</tr>
<tr>
<td></td>
<td>Oracle DB on Unix Pattern</td>
<td>cmdb_ci_db_ora_instance</td>
<td>$install_directory + &quot;/dbs*.ora&quot;</td>
</tr>
<tr>
<td>Tomcat</td>
<td>Tomcat</td>
<td>cmdb_ci_app_server_tomcat</td>
<td>$install_directory + &quot;/conf/server.xml&quot;</td>
</tr>
<tr>
<td>WMB</td>
<td>WMB On Unix Pattern</td>
<td>cmdb_ci_app_server_tomcat</td>
<td>$install_directory + &quot;/WEB-INF/web.xml&quot;</td>
</tr>
<tr>
<td></td>
<td>WMB On Windows Pattern</td>
<td>cmdb_ci_app_server_tomcat</td>
<td>$install_directory + &quot;/etc/config/<em>/</em>.prop&quot;</td>
</tr>
<tr>
<td>WMQ</td>
<td>WMQ On Windows Pattern</td>
<td>cmdb_ci_app_server_tomcat</td>
<td>$install_directory + &quot;/etc\config/<em>/</em>.prop&quot;</td>
</tr>
<tr>
<td></td>
<td>WMQ On Windows Pattern</td>
<td>cmdb_ci_app_server_tomcat</td>
<td>$install_directory + &quot;/bin/<em>/</em>.sh&quot;</td>
</tr>
</tbody>
</table>

## What to do

1. **Enable configuration file tracking by setting the glide.discovery.enable_file_tracking Discovery property to true.**

   You can also configure other properties to control the size and number of tracked configuration files, the time window during which changes to configuration files are tracked for a given version, and the number of changes allowed on a configuration file during that time window. See [Discovery properties](https://servicenow.com) for details.

2. **To prevent unauthorized access to the sensitive information from configuration files, perform the following steps:**

   a. Ensure that access control rules (ACL) are placed on the Tracked Configuration file table (cmdb_ci_config_file_tracked). Only allow authorized users to view this table or uncheck the Save Content setting.

   b. Ensuring that the tracked_file_reader role that controls access to configuration file data is assigned to correct users and user groups. By default, the itil role contains the
tracked_file_reader role giving all users with the itil role access to the configuration file information.

3. Verify that the Horizontal Discovery probe is active on the classifier for the software that you want to discovery. If not, you can enable it, specify the pattern, and then disable the other probes. See Add the Horizontal Pattern probe to a classifier for instructions.

4. Add or modify tracked file definitions to change the CI type or file path.

5. Set the tracked files deletion strategy to specify what you want to do with tracked configuration file CI records when pattern discovery can no longer find them.

6. Run horizontal discovery on the hosts that are running the applications you want to discover with patterns, open the application CI record, and check the Tracked Configuration Files related list.

7. Compare two versions of tracked CI configuration files to see the actual changes made to them.

Modify tracking changes in configuration files

Configure the system to collect information about changes in configuration files belonging to a configuration item (CI). Service Mapping uses this information to notify users that CI configuration files changed and to view actual changes to configuration files directly in the application service maps.

Role required: cloud_admin, sm_admin or admin

To enable tracking configuration files for a CI for which the system does track files, configure classification for the CI type to which this CI belongs:

- For CI types representing applications, create a Discovery process classification for the relevant pattern. Add the Horizontal Pattern probe in the Probe column, and then specify your pattern in the Pattern column.
- For CI types representing SNMP devices, perform configuration as described in Create a Discovery CI classification.

Note: There is no need to create CI classifications for hosts because these classifications are included in the base system.

The system tracks configuration files as part of horizontal discovery process of a CI to which these files belong. Configuration files contain CI settings and parameters. Service Mapping uses this information to notify users that CI configuration files changed and to view actual changes to configuration files directly in the application service maps. To learn about how the system tracks configuration files, stores and updates information about these files, see Configuration file tracking.

The system tracks configuration files for CI types with process classifiers that trigger patterns. The patterns contain tracked file definitions, which specify the CI type that the application belongs to and the path of the configuration file. Several patterns, such as the IIS pattern, provide tracked file definitions by default. You can use these default definitions or create your own for any pattern a process classifier triggers.

1. Navigate to Pattern Designer > Discovery Patterns.
2. Click the pattern for the relevant CI.

Attention: Enabling configuration file tracking does not modify the contents of a pattern.

3. Click the Tracked Files tab.
One or more tracked file definitions appear by default for Discovery patterns that support configuration file tracking.

If you do not see the Tracked Files tab, verify that the pattern is specified process classier. See Add the Horizontal Pattern probe to a classifier for instructions.

4. Click Edit to modify an existing definition or click New to create a new one.
5. Fill out or modify the form fields (see table):
Tracked File Definition

Define the file path for the selected CI type. File path consists of CI attributes in variable or wildcard format. For example, $install_directory+/conf/*.xml

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Select either the primary or related CI type to which the configuration file belongs.</td>
</tr>
<tr>
<td>File Path</td>
<td>Define the path of the configuration file to track. You can use pattern variables and wildcards. For example, for the following path: &lt;the folder with the &quot;install_directory&quot; parameter&gt;/conf/&lt;any xml file&gt;, enter $install_directory+/conf/*.xml</td>
</tr>
<tr>
<td>Save Content</td>
<td>Select this check box to make this file available for viewing and comparing directly the ServiceNow instance.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this check box to enable tracking of this configuration file.</td>
</tr>
</tbody>
</table>

Example tracked file definition for IIS Server discovery

6. Click Done.
7. If necessary, set the tracked files deletion strategy as described in Set the deletion strategy for tracked configuration files

Run discovery on the hosts that are running the applications you want to discover with patterns, open the application CI record, and check the Tracked Configuration Files related list.
If you are using Service Mapping, verify that it started collecting information about changes to configuration files:

1. Navigate to the application service map containing the CIs.
2. On the Changes tab, check that the list includes a record of the change you made. For example, that the configuration file was added, modified or deleted.
Data collection and discovery using Netflow

Service Mapping can perform discovery based on data collected using the Netflow protocol. Netflow is a protocol that Service Mapping can use to collect data about CIs and their connections along with Netstat and lsof commands.

Using the Netflow protocol for collecting data is one of the traffic-based discovery methods. Other methods deployed by Service Mapping are using netstat and lsof commands and the VPC Flow Logs. For more information, refer to Traffic-based discovery in Service Mapping.

In base systems, traffic-based discovery uses only TCP-related data collected with the help of the netstat and lsof commands. Discovery based on Netflow and VPC logs requires additional configuration. You can enrich your traffic-based discovery by configuring Service Mapping to use the Netflow protocol.

The component, which receives data in the Netflow format is the Netflow Collector. Its location depends on whether you configure data collection for testing purposes or standard operation:

For the test purposes

This setup results in half automated data collection flow, where Service Mapping imports data only if you manually copy it from the Netflow Collector. You place the Netflow Collector on a server inside your organization network. This must be a server different from the server hosting the MID Server. You configure and test this setup as described in Configure onetime data import using Netflow for testing purposes.

For standard operation

This setup results in fully automated data collection flow, where all involved components send, collect and analyze data automatically. You place the Netflow Collector on the same server as the MID Server inside your organization network. For instructions, see Configure data collection using Netflow.

Netflow-based discovery has the following flow:

1. The Netflow daemon runs and receives data from switches communicating with servers in the organization. The Netflow Collector writes received data from the Netflow daemon.

2. The server, hosting the Netflow collector, uses the Netflow nfdump utility to write the data into the nfdump output file. This file summarizes the raw data on all switches used for server communication.
3. In testing setups, where the Netflow Collector is located not on the same server as the MID Server, you may need to convert the nfdump into the gzip format. Then you must manually copy the raw data in the nfdump output file onto the MID Server.
4. The MID Server processes the raw data in the nfdump output file and places the processed information onto the ECC queue.
5. A sensor retrieves the processes data from the ECC queue and writes it into the Flow Connection (sa_flow_connection) table.

6. Whenever Service Mapping checks the ECC queue and receives information on a discovered CI, it checks these tables for any data on outbound connections related to the CI: the cmdb_tcp and sa_flow_connection tables. If these two tables contain unique data that patterns did not discover, Service Mapping enriches the information about the CI connections and adds them to the map.
Configure onetime data import using Netflow for testing purposes

Configure and test Service Mapping discovery process based on data collected using the Netflow protocol.

Learn about Traffic-based discovery in Service Mapping.

Role required: admin or sm_admin

In base systems, traffic-based discovery uses only TCP-related data collected with the help of the netstat and lsot commands. Discovery based on Netflow and VPC logs requires additional configuration. You can enrich your traffic-based discovery by configuring Service Mapping to use the Netflow protocol. For more information about the way Service Mapping to collect Netflow data, see Data collection and discovery using Netflow.

For testing purposes, install the Netflow Collector (nfdump) on a Unix server inside your organization. In this case, this Unix server should be different from the server hosting the MID Server server.

Configure the ServiceNow connector to trigger MID Server to collect the data from the flow log and processes it.

1. Download and install the Netflow collector (nfdump) on a Unix or Ubuntu server inside your organization.
   - For a Linux server, download, compile, and install the nfdump package. You can download the nfdump package from https://sourceforge.net/projects/nfdump/.
   - For an Ubuntu server, install the nfdump package without predownloading or compiling it. Open the command-line window and run the following command:
sudo apt-get install nfdump

For an Ubuntu server, if the `apt-get` command fails, predownload the nfdump package, save it locally and then install it. Open the command-line window and run the following commands:

```
sudo dpkg -i nfdump_1.6.15-3_i386.deb
sudo apt-get -f install
```

**Note:** The file name for the nfdump package has the following format: `nfdump_<version number> .deb`. In this example it is `nfdump_1.6.15-3_i386.deb`.

2. Configure the Netflow collector to save data for one day:
   a) Open the command-line window on the server hosting the Netflow collector.
   b) Create a cron job by using the following command:
      ```
crontab -e
      ```
   c) Enter the following command using the correct paths:
      ```
      */10 * * * * /usr/local/bin/nfexpire -e /data/nfdump -t 1d
      ```

3. Create a file with the nfdump data. For example, use the following command:
   ```
nfdump -q -m -R /data/nfdump/ -o extended -t 2016/07/06.07:00:00-2016/07/06.07:10:00 ©inet and proto tcp© >> /tmp/my_file
   ```

4. If the file is very large, you can compress it using the gzip format. Use the following command:
   ```
gzip /tmp/my_file
   ```

5. Copy the nfdump data file to the MID Server.

6. Configure Service Mapping to receive data collected by the Netflow collector:
   a) Navigate to Service Mapping > Administration > Flow Connectors.
   b) Click New.
   c) Click ndfdump file.
   d) On the ndfdump file page, configure parameters as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A descriptive name for the connector.</td>
</tr>
<tr>
<td>nfdump data path</td>
<td>The path to a location on the MID Server to which you saved the nfdump data</td>
</tr>
<tr>
<td>MID Server</td>
<td>The MID Server, onto which you copied the nfdump file.</td>
</tr>
<tr>
<td>Gzipped file</td>
<td>If you converted the nfdump file into the gzip format before saving it on</td>
</tr>
<tr>
<td></td>
<td>the MID Server, set this parameter to true to unzip it.</td>
</tr>
</tbody>
</table>

   e) Click Submit.

7. Verify that Service Mapping collects data using Netflow:
   a) On the nfdump file form, select the newly configured connector and click Run now to start the data collection flow and populate the Flow Connection (sa_flow_connection) table.
b) Navigate to **System Definitions > Tables**.

c) Click the Flow Connection (sa_flow_connection) table.

d) Under **Related Links**, click **Show List**.

e) Verify that the table contains data.

If you are satisfied with the results of the test, configure Netflow-based data collection as described in [Configure data collection using Netflow](#).

**Configure data collection using Netflow**

Enable Service Mapping to perform discovery based on data collected using the Netflow protocol. This setup results in fully automated data collection flow, where all involved components send, collect, and analyze data automatically.

Learn about [Traffic-based discovery in Service Mapping](#).

**Role required:** admin or sm_admin

In base systems, traffic-based discovery uses only TCP-related data collected with the help of the netstat and lsof commands. Discovery based on Netflow and VPC logs requires additional configuration. You can enrich your traffic-based discovery by configuring Service Mapping to use the Netflow protocol. For more information about the way Service Mapping uses Netflow, see [Data collection and discovery using Netflow](#).

Configure the ServiceNow Netflow connector to trigger the MID Server to collect the data from the Netflow flows and process them.

1. Install the nfdump package on a server hosting the MID Server in your organization:
   - For a Linux server, download, compile, and install the nfdump package. You can download the nfdump package from [https://sourceforge.net/projects/nfdump/](https://sourceforge.net/projects/nfdump/).
   - For an Ubuntu server, install the nfdump package without predownloading or compiling it.
     
     Open the command-line window and run the following command:
     
     ```
     sudo apt-get install nfdump
     ```
     
   - For an Ubuntu server, if the `apt-get` command fails, predownload the nfdump package, save it locally and then install it. Open the command-line window and run the following commands:
     
     ```
     sudo dpkg -i nfdump_1.6.15-3_i386.deb
     sudo apt-get -f install
     ```

   **Note:** The file name for the nfdump package has the following format: `nfdump_<version number>`. deb. In this example it is `nfdump_1.6.15-3_i386.deb`.

2. Configure the Netflow collector to save the nfdump file in the required directory.
   
   a) Open the `/etc/init.d/nfdump` file.
   
   b) Modify the parameter responsible for saving this file in the required location.
     
     For example, on an Ubuntu server, specify the location using the `DEAMON_ARGS` parameter:
     
     ```
     DATA_BASE_DIR="/var/cache/nfdump"
     DAEMON_ARGS="-D -l $DATA_BASE_DIR -P $PIDFILE"
     ```

     For operational information, refer to [https://sourceforge.net/projects/nfdump/](https://sourceforge.net/projects/nfdump/).
3. Configure the switches to forward their nfdump files to the MID Server. The default value for the MID Server is port 9995.

4. Configure the Netflow collector to save data for one day:
   a) Open the command-line window on the server hosting the Netflow collector.
   b) Create a cron job by using the following command:
      ```
      crontab -e
      ```
   c) Enter the following command using the correct paths:
      ```
      */10 * * * * /usr/local/bin/nfexpire -e /data/nfdump -t 1d
      ```

5. Verify that the Netflow collector is configured correctly and receives the correct data from the network resources.
   a) Run the following command:
      ```
      nfdump -q -O tstart -R /data/nfdump/ -o extended
      ```
   b) In the command output, verify that marked fields contain real data:

6. Configure Service Mapping to receive data collected by the Netflow collector:
   a) Navigate to Service Mapping > Administration > Flow Connectors.
   b) Click New.
   c) Click nfdump install.
   d) On the nfdump install page, configure parameters as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A descriptive name for the connector.</td>
</tr>
<tr>
<td>MID Server</td>
<td>The MID Server on which you installed the Netflow collector.</td>
</tr>
<tr>
<td>nfdump data directory</td>
<td>The data directory where you configured the Netflow collector to save the nfdump files.</td>
</tr>
</tbody>
</table>

   e) Click Submit.

7. Verify that Service Mapping collects data using Netflow:
   a) On the nfdump install form, select the newly configured connector and click Run now to start the data collection flow and populate the Flow Connection (sa_flow_connection) table.
   b) Navigate to System Definitions > Tables.
   c) Click the Flow Connection (sa_flow_connection) table.
   d) Under Related Links, click Show List.
e) Verify that the table contains data.

**Data collection and discovery using VPC Flow Logs**

Service Mapping can perform discovery based on data collected using VPC Flow Logs. Amazon VPC hosts Amazon Elastic Compute Cloud (EC2) instances that provide Amazon Web Services. VPC Flow Logs collect data on IP traffic going to and from network interfaces in the VPC.

In base systems, traffic-based discovery uses only TCP-related data collected with the help of the `netstat` and `lsof` commands. Discovery based on Netflow and VPC logs requires additional configuration. You can enrich your traffic-based discovery by configuring Service Mapping to use VPC Flow Logs.

Service Mapping discovery based on VPC Flow Logs has the following flow:

1. Amazon EC2 instances collect their individual logs into log streams and forward them to the central flow log group.

2. The ServiceNow connector triggers MID Server to collect the data from the flow log and processes it.

3. The MID Server places the processed information onto the ECC queue.
4. A sensor retrieves the processes data from the ECC queue and writes it into the Flow Connection (sa_flow_connection) table.

5. Whenever Service Mapping checks the ECC queue and receives information on a discovered CI, it checks these tables for any data on outbound connections related to the CI: the cmdb_tcp and sa_flow_connection tables. If these two tables contain unique data that patterns did not discover, Service Mapping enriches the information about the CI connections and adds them to the map.
In deployments with multiple flow log groups, configure a dedicated connector that works with one MID Server for every flow log group. Multiple flow log groups may use the same AWS credentials.

**Configure data collection using VPC Flow Logs**
Enable Service Mapping to perform discovery based on data collected using Virtual Private Cloud (VPC) logs. This method is relevant for organizations using Amazon Web Services (AWS).

Verify that credentials for the AWS account are configured in the Credentials module of the platform by navigating to **Service Mapping > Credentials > AWS Credentials**. For more information about AWS credentials, see [Cloud credentials](#).

Role required: admin or sm_admin

In base systems, traffic-based discovery uses only TCP-related data collected with the help of the `netstat` and `lsof` commands. Discovery based on Netflow and VPC logs requires additional configuration. You can enrich your traffic-based discovery by configuring Service Mapping to use VPC Flow Logs. For more information about the Service Mapping discovery flow based on VPC Flow logs, see [Data collection and discovery using VPC Flow Logs](#).

Amazon VPC hosts Amazon Elastic Compute Cloud (EC2) instances that provide Amazon Web Services. VPC flow logs collect data on IP traffic going to and from network interfaces in the VPC.

Configure the ServiceNow connector to trigger MID Server to collect the data from the flow log and processes it. In deployments with multiple flow log groups, configure a dedicated connector that works with one MID Server for every flow log group. Multiple flow log groups may use the same AWS credentials.

2. Configure Service Mapping to work with VPC Flow Logs:
   a) Navigate to **Service Mapping > Administration > Flow Connectors**.
b) Click New.

c) Click AWS VPC flow logs.

d) On the AWS VPC flow logs page, configure the connector parameters as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A descriptive name for the connector.</td>
</tr>
<tr>
<td>Group name</td>
<td>The name of the central flow log group to which Amazon EC2 instances forward their log streams.</td>
</tr>
<tr>
<td>MID Server</td>
<td>The MID Server that Service Mapping uses to collect the data from the flow log group on AWS.</td>
</tr>
<tr>
<td>AWS Credentials</td>
<td>Select AWS credentials relevant for the account for which you want to collect flow logs.</td>
</tr>
</tbody>
</table>

e) Click Submit.

3. Verify that Service Mapping collects data using VPC Flow Logs:

a) On the AWS VPC flow logs form, select the newly configured connector and click Run now to start the data collection flow and populate the Flow Connection (sa_flow_connection) table.

b) Navigate to System Definitions > Tables.

c) Click the Flow Connection (sa_flow_connection) table.

d) Under Related Links, click Show List.

e) Verify that the table contains data.

Configure Search Assistant for Windows

The Search Assistant feature of Pattern Designer allows you to search within files or registries. Upload grep files on to your instance to enable this feature to search on Windows servers.

Role required: admin or sm_admin

Grep is a utility that the Search Assistant feature uses to search files for matches. Grep is native to Unix-based systems, therefore you must upload it only for Windows-based configuration items (CI). Once you upload grep files onto the instance, it transfers them onto all MID Server which distribute the files onto all Windows-based CIs in your organization.

1. Download the following files onto the computer from the Internet.

   - egrep.exe
   - libiconv2.dll
   - libintl3.dll
   - pcre3.dll
   - regex2.dll

   Note: Make sure to unzip folders if necessary.

For example, you can download the zip files for the Dependencies and Binaries folders at the following location: [http://gnuwin32.sourceforge.net/packages/grep.htm](http://gnuwin32.sourceforge.net/packages/grep.htm)
After you unzip the downloaded folders, the files are located as follows:

- grep-2.5.4-bin/bin/egrep.exe
- grep-2.5.4-dep/bin/libiconv2.dll
- grep-2.5.4-dep/bin/libintl3.dll
- grep-2.5.4-dep/bin/pcre3.dll
- grep-2.5.4-dep/bin/regex2.dll

2. Navigate to Pattern Designer > Uploaded Files.
3. Click New.
4. Enter the original file name in the Name field on the Logical Name tab, for example egrep.exe.
5. Click OS Types and select Windows.
6. Click **OS Architectures** and select both 32-bit and 64-bit option.
7. Click the **Manage Attachments** icon.
8. Click **Choose Files**.
9. In the **Downloads** folder on your computer, navigate to the file which name you entered before, for example **egrep.exe**.
10. Click **Choose**.
11. Close the Attachments window.
12. Repeat **Step 3 to Step 11** to upload the rest of the files:
   - **libiconv2.dll**
Modify display for CI attributes

You can control what configuration item (CI) attributes the system displays in the Properties pane of application service maps.

Get familiar with the notion of views. For more information on views, see View management.

Decide for which CI classes you need to modify display.

Decide whether you want to modify display for parent CI classes to see the view change in all their child CI classes. To check the parent CI class of a CI class, perform the steps below:

Note: Service Mapping user interface refers to CI classes as CI types.

1. Navigate to System Definition > Tables.
2. Set the search field to Name, and enter the relevant CI class.
   The list of all CI class that match the name is displayed.
3. Check the Extends table column to see the parent CI class.

Role required: sm_admin or app_service_admin

When working with application service maps, the Properties pane displays attributes for CIs and connections you select in a map. Since there is a large number of attributes to display, you may want to choose which attributes are displayed, and in what order.

The attributes and their order of display is determined by a combination of the CI type definition and the view for this CI type.

CMDB stores CI class in the form of a hierarchy where some CI classes are parents to other CI classes, who automatically derive their parent's attributes in addition to attributes you configure specifically for child CI classes. CI class hierarchy is used widely for configuring CI behavior, relationships, and display.
The view that controls the attribute display in application service maps is called **sa_map_properties**. You can assign this view to new CI classes or modify the view for CI classes which already use it. Use the CI type hierarchy to configure the **sa_map_properties view**:

- The parent CI class at the top of the hierarchy is **cmdb**.
- If you do not define the view for a child CI class, the system displays attributes derived from the parent with the addition to all the attributes of this child CI class.
If you define a view for a child CI class, it includes only the attributes that you specifically added to the child CI class. The system does not display any of its parent attributes automatically.
• If you do not add a parent CI class or its child CI class to the view, the system uses the ascendant CI class that was added to the view in addition to all attributes of all parent CI classes in the hierarchy between this child CI classes and the parent CI class that is added to the view.
For example, if you define the `sa_map_properties` view for the CI class for Windows Servers, the Properties pane displays the same attributes for all Windows Servers: CI type (class), Class, Model ID, and name.
You can modify the view at any time.

1. Navigate to **Configuration > Application Services > Application Services**.
2. Alternatively, if Service Mapping is deployed, navigate to **Service Mapping > Services > Application Services**.
3. Click **View map** next to the application service that contains the required CIs.
4. In the application service map, select the CI whose attributes you want to change.
5. In the **Properties** pane, click **Detailed Properties**.
6. In the CI Detailed Properties page, click the **Additional actions** icon and select **Configure > Form Layout**.

7. In the **View name** list under **For view and section**, define which view you want to modify:

   - Select **sa_map_properties** if it appears in the list, or
   - If **sa_map_properties** does not appear in the list:
     1. Select **New**.
     2. In the **View name** field, enter **sa_map_properties**.
     3. Click **OK**.

8. Use the slushbucket to add or remove attributes from the view.

9. Click **Save**.

10. Return to the application service map and reselect the CI for which you changed the view.
The Properties pane displays the attributes that you configured for the view.

11. If necessary, repeat the procedure for other CI classes.

Configure support for Windows servers with non-English OS

You can configure your Now Platform to support Windows servers that use non-English Windows operating system (OS).

You can perform this procedure only on the MID Servers that serve locations using non-English operating system. Alternatively, you can perform this configuration on all MID Servers in one go, even MID Servers that are used for locations using English operating systems.

Role required: mid_server

If your organization does not deploy Service Mapping, this configuration is not necessary.

Sometimes organizations choose to use operating systems in local languages. While having a localized operating system (OS) may be user-friendly, it poses a problem when it comes to automatically discovering Windows servers running the localized OS. During the discovery and mapping process Service Mapping sends discovery commands to Windows servers on your infrastructure via a MID Server. If a Windows server returns a message in a language other than English, Service Mapping does not recognize it as a valid response and the interaction fails.

If your organization uses non-English operating system for some of the Windows servers, you must perform this procedure to make sure that Service Mapping can access all Windows servers, not exclusively Windows servers using the English operating system. This configuration allows the MID Server, that is located between Service Mapping and Windows servers, to recognize a non-English response from a Windows server and to change the language of the CI operating system into English.

This configuration affects only the user for which you perform this procedure.

1. Navigate to MID Servers > Properties in the Modules pane.
2. Click New.
3. In the Name field, enter mid.servicewatch.wmi.mui.
4. In the Value field, enter true.
5. In the MID server field, select the server to which you want to apply this change:

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>To apply the change to all MID Servers in your organization</td>
<td>Leave the MID Server field blank.</td>
</tr>
<tr>
<td>To apply the change to a specific MID Server</td>
<td>Select the relevant MID Server from the list.</td>
</tr>
</tbody>
</table>

6. Click Submit.

Upload the rctrlx.exe file to MID Servers


Role required: admin or sm_admin

During the discovery process, the MID Server runs Service Mapping and Discovery patterns on applications and devices located inside the organization network. A pattern is a sequence of steps whose purpose is to detect attributes of a CI and its outbound connections. To execute commands over PowerShell correctly, MID Server runs patterns as services using the rctrlx.exe file. The rctrlx.exe file is part of the open source ManagePC utility and is available for downloading from the Internet.
Upload an `rctrlx.exe` file for every MID Server in your deployment.

1. Download the `rctrlx.exe` file onto your computer from the Internet.
   For example, [https://github.com/leonsodhi/rctrlx/releases](https://github.com/leonsodhi/rctrlx/releases) offers the latest versions of the `rctrlx.exe` file:

2. Place the `rctrlx.exe` file into the folder for Windows 32-bit servers:
   `<MID Server installation directory>\agent\bin\sw_wmi\bin\32`
   For example, `C:\SN_MID\SN_MID_Dev\agent\bin\sw_wmi\bin\32`.

3. Place the `rctrlx.exe` file into the folder for Windows 64-bit servers:
   `<MID Server installation directory>\agent\bin\sw_wmi\bin\64`
   For example, `C:\SN_MID\SN_MID_Dev\agent\bin\sw_wmi\bin\64`.

**Plan application services**

You can centrally manage the process of planning, creating, and reviewing numerous application services using the Service Map Planner.

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see [KB0689681: Features replacing deprecated Service Planner in Service Mapping](https://support.servicenow.com/index.html). Service Map Planner module refers to application services as application services.

Typically, the process of creating an application service is intensive. It includes collecting information and performing multiple tests until you get a satisfying result. Most organizations have hundreds or thousands of services which makes this process even harder to manage. Service Mapping allows you to gather and test all the information related to application services in one central location.

You can create reports based on planning phases to monitor progress you made in planning and creating application services.

The process of planning and creating application services using Service Map Planner consists of several stages:

1. Decide how you want to manage your application service planning by splitting application services into groups. These groups are referred to as phases. You are free to organize application services into phases whichever way you find effective: by location, by type of service, by priority, and so on. For example, if you decided to start your planning project from application services in the EMEA office, you create a phase with that name and populate it with all application services belonging to this office.

   You do not have to accomplish planning phases in Service Mapping in any particular order. On the contrary, you can simultaneously make progress on testing and reviewing application services belonging to different phases.

2. To begin the actual work on creating application services, you must add planned application services to phases. It creates an association between application services and their phases, which helps to manage creating application services. In our example, you add all application services used by the EMEA office to the EMEA phase.

3. **Creation of actual application services during planning**

   You create each application service individually even if you simultaneously added multiple application services to the phase. During this stage, an administrator responsible for planning application services collects data and runs the mapping process. Then a staff member who is familiar with the infrastructure and applications reviews mapping results for correctness and approves the newly created application service.
Going through these actions may take a long time. As a rule, a phase contains planned application services at different stages depending on how much progress you were able to make.

**Planning flow for application services**

1. **Decide on planning strategy and create phases**
2. **Populate phases**
3. **Create individual services**:
   - Collect data
   - Test connectivity
   - Create and map service
   - Review and approve service

**Create empty phases**

Service Mapping allows you to split planned application services into groups which are referred to as phases.

Role required: sm_admin or sm_app_owner

Set up Service Mapping as described in [Service Mapping setup](#).

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see [KB0689681: Features replacing deprecated Service Planner in Service Mapping](#).

Decide how you want to manage your application service planning by splitting application services into groups. These groups are referred to as phases. Then populate phases with application services. You are free to organize application services into phases whichever way you find effective: by location, by type of service, by priority, and so on. For example, if you decided to start your planning project from application services in the EMEA office, you create a phase with that name and populate it with all application services belonging to this office.
You do not have to accomplish planning phases in Service Mapping in any particular order. On the contrary, you can simultaneously make progress on testing and reviewing application services belonging to different phases.

1. Navigate to Service Mapping > Service Map Planner > Phases.
2. Click New.
3. Define phase attributes as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive, but short name for the phase.</td>
</tr>
<tr>
<td>Description</td>
<td>Provide a short description of the phase.</td>
</tr>
<tr>
<td>Status</td>
<td>Verify that the status is set to Active.</td>
</tr>
</tbody>
</table>

4. Click Submit.
   The new phase appears in the list of phases.

**Phase population**

You associate planned application services to phases.

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see [KB0689681: Features replacing deprecated Service Planner in Service Mapping](#).

To begin the actual work on creating application services, you must add planned application services to phases. It creates an association between application services and their phases, which helps to manage creating application services. In our example, you add all application services used by the EMEA office to the EMEA phase.

You can associate an application service using different methods depending on how you have been managing your application services up until now and what information about them you gathered.

### Methods of associating planned application services to phases

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import from the CSV file</td>
<td>This method suits you if your organization has performed cross-organization mapping and analysis and collected some information about planned application services. If so, you can organize the collected information in a specific order and save it as a CSV file. Service Mapping extracts information from this file and creates potential application services referred to as service candidates.</td>
<td>You associate multiple application services at a time. There is a relatively low chance of mistakes.</td>
<td>You must prepare all the necessary information in a very specific format before you can perform the import.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
<td>Advantages</td>
<td>Disadvantages</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Import from a load balancer</td>
<td>To enable you to map your application services in bulk, Service Mapping extracts entries directly from load balancers and convert them into application service candidates.</td>
<td>You associate multiple application services at a time. There is no preparation.</td>
<td>The result of the import is not precise. There may be many false or duplicate application services created by mistake because the data from the load balancer is raw.</td>
</tr>
<tr>
<td>Add application services manually</td>
<td>Use this method if you do not have any organized data on your application services or you do not want to use other methods.</td>
<td>There is no preparation. There is a relatively low chance of mistakes.</td>
<td>If you have many application services for a phase, the process may take a long time.</td>
</tr>
</tbody>
</table>

**Import planned application services from the CSV file**

Associate multiple application services to a phase in a single step by importing them from the CSV file containing necessary information.

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.

Prepare the information necessary for the import in the CSV file:

1. Copy the following template in the .xlsx format: importCSVexample.xlsx.
2. Populate the fields in the .xlsx file:

   **Warning:** Do not modify column labels in the .xlsx file.

   **Warning:** If information on application services is not formatted in the .xlsx file according to the template, Service Mapping fails to create planned services or creates them with errors.

   **Warning:** This .xlsx format is correct only for importing business services using the Service Map Planner module. When you map business services from a CSV file from the new Home page, follow the procedure in Prepare a CSV file for mapping your candidates.

<table>
<thead>
<tr>
<th>Column label</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the planned application service.</td>
</tr>
<tr>
<td>ServiceDeploymentOwner</td>
<td>Enter the service deployment owner name as it is displayed in the platform: <code>&lt;firstname.lastname&gt;</code>. The service deployment owner is responsible for creating and fixing application services during planning. The service deployment owner must have the sm_admin role.</td>
</tr>
<tr>
<td>Column label</td>
<td>Information</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ApplicationOwners</td>
<td>Enter the application owner owner. The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Use the following syntax: <code>&lt;firstname.lastname&gt;</code>. To enter more than one name, use semicolon (:) to separate names.</td>
</tr>
<tr>
<td>PlannedEntryPoints</td>
<td>Enter the HTTPS entry points that Service Mapping uses to discover this application service. To enter more than one entry point, use semicolon (:) to separate them. An entry point is how clients access an application service. Usually, it is either a URL or a combination of the IP address and port. Service Mapping starts the mapping process from this point. For example, to map your electronic mailing application service, define an IP address or host name of the email server as an entry point.</td>
</tr>
<tr>
<td>OtherEntryPoints</td>
<td>Enter information about entry points that either a service deployment owner or application owner manually defines for this application service. Enter this information as free text.</td>
</tr>
<tr>
<td>ComponentsDetails</td>
<td>Enter information about components you expect to discover as part of this application service. Enter this information as free text.</td>
</tr>
</tbody>
</table>

3. Save the file in a drive that you can access during the import with the `.csv` extension.

**Role required: sm_admin**

This method suits you if your organization has performed cross-organization mapping and analysis and collected some information about planned application services. If so, you can organize the collected information in a specific order and save it as a CSV file. Service Mapping extracts information from this file and creates potential application services referred to as service candidates.

1. Navigate to **Service Mapping > Service Map Planner > Phases**.
2. Click the relevant phase.
3. On the Phase page, click **Import from CSV**.
4. In the Select CSV file to import window, click **Choose file**.
5. Navigate to the CSV file to use for the import and click **Choose**.
6. Click **Import**.
   The newly associated application services appear under Business Service Planning.
7. If necessary, troubleshoot import-related errors.
   a) Click the **Import Errors** tab.
   b) Review information on application services that Service Mapping failed to import.
   c) Create a separate CSV file to these application services.
**Note:** Make sure that the information and its format are correct.

d) Import application services from the second CSV file by repeating 3-6.

*Import planned application services from load-balancers*

Associate multiple application services to a phase in one go by importing them from load balancers.

Role required: sm_admin

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.

To enable you to map your application services in bulk, Service Mapping extracts entries directly from load balancers and convert them into application service candidates.

1. Navigate to **Service Mapping > Service Map Planner > Phases**.
2. Click the relevant phase.
3. On the Phase page, click **Import from load balancer**.
   The Load Balancer Services page shows the list of entries in all load balancers configured in the instance.
4. Mark the entries that you want Service Mapping to import as a application service by clicking check boxes next to them.
5. Click **Import**.
   The newly associated application services appear under Business Service Planning.

   By default, Service Mapping assigns the following parameters for every application service:
   - **Name** - based on the following rule: <load balancer name>+<VIP name>. You can change names at a later stage.
   - **Entry point type** - inferred from the port value as follows:

<table>
<thead>
<tr>
<th>Port number</th>
<th>Entry point type</th>
</tr>
</thead>
<tbody>
<tr>
<td>443</td>
<td>HTTPS</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
</tr>
<tr>
<td>Any number except 443 and 80</td>
<td>TCP</td>
</tr>
</tbody>
</table>

6. If necessary, troubleshoot import-related errors.
   a) Click the **Import Errors** tab.
   b) Review error messages and fix issues that caused errors.

*Add planned application services to phases manually*

If you cannot associate an application service to a phase by importing it, add it manually.

You can add an application service only to an existing phase.

Role required: sm_admin

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.
Use this method if you do not have any organized data on your application services or you do not want to use other methods.

1. Navigate to **Service Mapping > Service Map Planner > Phases**.
2. Click the relevant phase.
3. On the Phase page, click **New**.
4. Define attributes for the business service you want to add:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a descriptive and short name for this application service.</td>
</tr>
<tr>
<td>Application Owner</td>
<td>The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. The owner cannot create or modify application services. Click the <strong>Add me</strong> or <strong>Edit Application Owner</strong> icons to define this parameter.</td>
</tr>
<tr>
<td>Service Deployment Owner</td>
<td>The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. The owner cannot create or modify application services. Select the name from the list. If you add more than one name, separate them with a semicolon (;).</td>
</tr>
</tbody>
</table>

5. Click **Submit**.

**Creation of actual application services during planning**

You create each application service individually even if you simultaneously added multiple application services to the phase.

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see **KB0689681: Features replacing deprecated Service Planner in Service Mapping**.

After you created phases and populated them with application services, you are ready to look at individual application service in a phase. Application services in phases are only planned services, they are drafts. You are going to create actual application services based on them. The actual application service inherits all information defined for the planned service. This new actual application service appears in the list of all application services and has the Non-Operational state.

The following users collaborate to create actual application services:
• Service deployment owner

Ideally, only the service deployment owner should create the actual application service, however, any user with the sm_admin role can perform this task.

• Application owner

The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. The owner cannot create or modify application services.

Notice that although service deployment owner fixes the actual application service, the application owner provides feedback and approves the draft version of it - the planned application service. It allows you to have all the information related to planning and creation of the application service in one central place.

Creating an actual application service during planning has the following flow:

Collect data
A service deployment owner or an application owner collects data for the planned application service.

Role required: sm_admin or sm_app_owner

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.

At this stage you are planning your application service and collect information that Service Mapping uses later to create, review, and approve this application service.

You provide information on people responsible for creating and reviewing your application service.

Make sure that your planned application service has some entry points assigned to it. An entry point is how clients access an application service. Service Mapping starts the mapping process from this point. You cannot create an actual application service, review, and approve the planned application service unless it has at least one entry point. Even if this planned application service has some entry points assigned to it during import, you can add additional entry points if necessary.

Some information, such as information about components, is not used to create an application service, but to review it at a later stage.

1. Navigate to Service Mapping > Service Map Planner > Phases.
2. Select the phase which contains the application service.
3. Click the name of the application service.
4. Define the application owner for this application service.

The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. The owner cannot create or modify application services.

a) Click the Add me icon next to Application Owner to define yourself as the application owner.

Or
b) Click **Edit Application Owner** (Padlock) icon next to **Application Owner** and enter a single or multiple names. If necessary, enter an email.

5. Define the service deployment owner for this application service in the **Service Deployment Owner** field.

The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. The owner cannot create or modify application services.

Select the name from the list. Separate multiple names with a semicolon (;).

6. If there is no entry point, add it.

An entry point is how clients access an application service. Usually, it is either a URL or a combination of the IP address and port. Service Mapping starts the mapping process from this point. For example, to map your electronic mailing application service, define an IP address or host name of the email server as an entry point.

   a) Click **New** on the **Planned Entry Points** tab.

   b) Select the relevant type for the entry point from **Entry point type**.

      If there is no entry point type you need, enter its description on the **Other Entry Points** tab.

   c) Define other entry point attributes that vary depending on the type you selected.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host/Hostname/Host Name</td>
<td>The value of the target server on which the service is running. This value can contain a real host name, alias, IP, or VIP.</td>
</tr>
<tr>
<td>Port</td>
<td>The port number of the service that you want to discover.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the service that you want to discover.</td>
</tr>
<tr>
<td>URL</td>
<td>The URL of the service that you want to discover.</td>
</tr>
</tbody>
</table>

   d) Click **Submit**.

7. Provide information for components that are part of this application service, and that you know of.

   a) Click the **Components** tab.

   b) Click **New**.

   c) Enter description of the application service component.

   d) Click **OK**.

---

**Check entry point connectivity**

A service deployment owner or an application owner tests that Service Mapping can connect to the devices that are expected to make up this application service. This feature is not available on instances using Edge Encryption.

Run either horizontal or top-down discovery on the host for which you want to check connectivity.

Role required: sm_admin or sm_app_owner
Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.

You can check if Service Mapping can connect to hosts to identify any connectivity-related issues. You can investigate the problem source, fix it, and run the connectivity check again to verify that you are ready to create an actual application service. The most common causes of connectivity problems are missing or wrong credentials for a host.

You can check connectivity of entry points you defined for a planned application service. An entry point is how clients access an application service. Service Mapping starts the mapping process from this point. You can also check connectivity of hosts connected to specified entry points.

As part of defining entry points for an application service, you enter their IP addresses. When you run this test, Service Mapping checks the corresponding record in the CMDB (cmdb_ci) for information on credentials for IP addresses you defined. By default, Service Mapping checks the IP address that you defined for the entry point, but you can add additional IP addresses during the procedure.

To check connectivity of hosts connected to entry point IP addresses, you must specify the number of levels to check as Hop Count. Service Mapping uses information on TCP and Netflow connections stored in the CMDB tables, to find hosts connected to the IP address.

### Tables containing data collected using traffic-based methods

<table>
<thead>
<tr>
<th>Table</th>
<th>Source</th>
<th>Used by Service Mapping to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow Connection (sa_flow_connection)</td>
<td>Netflow and VCP logs</td>
<td>Discover dependencies, add connections during top-down discovery.</td>
</tr>
<tr>
<td>TCP Connection (cmdb_tcp)</td>
<td>netstat and lsof commands</td>
<td>Discover connections during top-down discovery.</td>
</tr>
</tbody>
</table>

1. Navigate to Service Mapping > Service Map Planner > Phases.
2. Select the phase which contains the planned application service.
3. Click the name of the planned application service.
4. On the Planned Entry Points tab, click the check boxes next to the entry points whose credentials you want to check.
5. Click Check Connectivity.

The Check IP Address Connectivity window opens displaying the IP address you defined for the entry points.
6. If necessary, modify or add IP addresses in the IP Address(es) field.

   Use comma without whitespace to separate multiple IP addresses.
7. Select the number of levels to check in the Hop Count list.
   The number of hops performed by the actual test depends on the ability of Service Mapping to connect to hosts in the next level. If you use Netflow to collect data, you can see CIs on all levels even if Service Mapping cannot connect to them.
8. Click OK.

   The Connectivity Test page opens.
9. Verify that the status of the test is Completed.
10. Review the result of the test:
    - If the status and the message state that the IP address is found, Service Mapping can connect to this host.
If the status is Info, there is a possibility that there is a connectivity problem or that this host is not in the CMDB.

If the status is Not found, either there is a connectivity problem or this host is not in the CMDB.

11. Click the IP address to read the full message. Use this information to troubleshoot connectivity issues.

To troubleshoot, you may need to:
- Run discovery on a problematic CI either from Discovery or Service Mapping.
- Define missing or wrong credentials for this CI.

12. Perform this procedure on the same entry points to verify that Service Mapping does not experience the same connectivity issues.

Create actual application services during planning

The service deployment owner creates an actual application service based on the planned one.

Role required: sm_admin

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.

The actual application service inherits all information defined for the planned service. This new actual application service appears in the list of all application services and has the Non-Operational state.

Ideally, only the service deployment owner should create the actual application service, however, any user with the sm_admin role can perform this task.
When Service Mapping creates an actual application service during planning, it performs the following validation:

- Checks that the entry point is not used for another application service.
- Checks that the application service name is unique and does not exist in the CMDB.

1. Navigate to **Service Mapping > Service Map Planner > Phases**.
2. Select the phase which contains the planned application service.
3. Click the name of the planned application service.
4. Make sure that the name and the service deployment owner are defined.
5. Click **Create and Discover**.

   Service Mapping creates the actual application service and starts the process of discovery and mapping.

   When the discovery process completes, the **View map** link appears on the page.

6. If necessary, troubleshoot validation errors:

   a) Click the **Validation Errors** tab.
   b) Review error messages and fix the issues that caused them.

7. Enter the name of the application owner. The application service owner is a user who is familiar with the infrastructure and applications making up the service. This user is the application service SME who provides information necessary for a successful creation of an application service.

8. Click **Send for Review**.

   At this stage, you let the application owner review mapping results and provide feedback in the form of notes on the planned application service page.

---

**Review application services**

An application owner reviews mapping results for correctness and provides feedback for the application service.

Role required: sm_app_owner

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see [KB0689681: Features replacing deprecated Service Planner in Service Mapping](https://service-now.com/kb).  

As a staff member that is familiar with the infrastructure and applications, review mapping results of the actual application service and provide your feedback in the form of notes. After that, a service deployment owner reviews and implements your feedback, and possibly, leaves some notes for you. The process of application service review may take some time as it requires making changes and running the mapping process on the application service. Typically, it takes several iterations to arrive at the desired result. Once the service deployment owner fixes all issues, you can approve it.

Ideally, only the application owner reviews the planned application service, however, any user with the sm_app_owner role can perform this task. As the application owner for the planned application service, you receive an email notification that the application service is assigned to you for review.

1. Navigate to the planned application service:
   a) Navigate to **Service Mapping > Service Map Planner > Business Service Planning**.
   b) Sort the list of planned application services by status and scroll to see services in In Review status.

   Or
Use the filter to narrow the list.

c) Click the required application service.

2. Review the actual application service:
   a) On the Planned Business Service page, check entries on the Components tab to see which devices and applications are expected to be part of this application service.
   b) Click View Map and review the map.
   c) Return to the planned application service, enter your comments for the service deployment owner in the Notes field and click Post.
      At this stage, let the service deployment owner read your comments and make the necessary changes.

3. Approve the planned application service:
   a) Navigate to the same application service.
   b) Review the notes from the service deployment owner.
   c) Click View Map and check that the changes you required are implemented.
   d) If the mapping results are satisfactory, click Approve.

Fix actual application services during planning
The service deployment owner checks the feedback and updates the actual application service as necessary.

Role required: sm_admin

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see KB0689681: Features replacing deprecated Service Planner in Service Mapping.

The process of application service review may take some time as it requires making changes and running the mapping process on the application service. Typically, it takes several iterations to arrive at the desired result.

Ideally, only the service deployment owner should fix the actual application service, however, any user with the sm_admin role can perform this task.

1. Navigate to Service Mapping > Best Practice > Planned Business Services.
2. Sort the list by status.
   Business services reviewed by the application owner are in In Review status.
3. Click one of the application services in In Review status.
4. Review the notes made by the application owner.
5. Click View map.
   The map of the actual application service opens.
6. Make necessary changes on the actual application service page.
7. Check the map for correctness.
8. Enter your comments for the application owner in the Notes field and click Post.
   At this stage, let the application owner read your comments and review the changes you made.

Approve application services during planning
The application owner reviews the fixed application service and approves it if the result is satisfactory.
Role required: sm_app_owner

Fresh install deployments do not have the Service Map Planner module. The improved planning functionality is provided as an integral part of the new Service Mapping user interface. For more information, see [KB0689681: Features replacing deprecated Service Planner in Service Mapping](#).

As a staff member that is familiar with the infrastructure and applications, review the mapping results of the actual application service and provide your feedback in the form of notes. Then, a service deployment owner reviews and implements your feedback, and possibly, leaves some notes for you. The process of application service review may take some time as it requires making changes and running the mapping process on the application service. Typically, it takes several iterations to arrive at the desired result. Once the service deployment owner fixes all issues, you can approve the application services.

Ideally, only the application owner should approve the planned application service, however, any user with the sm_app_owner role can perform this task.

1. Navigate to the planned application service:
   a) Navigate to Service Mapping > Best Practice > Planned Business Services.
   b) Sort the list of planned application services by status and scroll to see services in In Review status.
      Or
      Use the filter to narrow the list. For example, filter by your name if you are the application owner.
   c) Click the required application service.

2. Review notes made by the service deployment owner.
3. Click View Map and check that the changes you required are implemented.
4. If the mapping results are satisfactory, click Approve.

### Discovery

ServiceNow® Discovery finds applications and devices on your network, and then updates the CMDB with the information it finds.

Discovery is available as a separate subscription from the rest of the Now Platform. See [Request Discovery](#) for details.

**Explore**
- Upgrade to New York
- Discovery basics
- Data collected by Discovery
- Domain Separation and Discovery

**Set up**
- Request Discovery
- Installing the MID Server
- Create and test your credentials
- Discovery setup

**Administer**
- Discovery properties
- Create a Discovery behavior
- Create a Discovery CI classification
- Create a Discovery process classification

**Use**
- Schedule a horizontal discovery
- Run a Quick Discovery
- Discovery status
- Validate discovery results

**Integrate and develop**
- Activate SCCM Asset Intelligence scheduled imports
- Discovery APIs on the developer portal

**Troubleshoot and get help**
- Discovery monitoring and issue resolution
- Search the HI knowledge base for known error articles
- Ask or answer questions in the ITOM forum

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Discovery basics

Discovery finds computers, servers, printers, a variety of IP-enabled devices, and the applications that run on them. It can then update the CIs in your CMDB with the data it collects.

Horizontal discovery and top-down discovery

There are two types of discovery:

Horizontal discovery

Horizontal discovery is a technique that Discovery uses to scan your network, find computers and devices, and then populate the CMDB with the CIs it finds. Horizontal discovery does create direct relationships between CIs, such as a runs on relationship between an application CI and the actual computer CI that it runs on. Horizontal discovery is not aware of business services and does not create relationships between CIs based on the business service they are in.

Top-down discovery

Top-down discovery is a technique that Service Mapping uses to find and maps CIs that are part of business services, such as an email service. For example, top-down discovery can map a website business service by showing the relationships between an Apache Tomcat web server service, a Windows server, and the MSSQL database that stores the data for the business service.

Typically, Service Mapping and Discovery work together to run horizontal discovery first to find CIs, and then top-down discovery to establish the relationships between business services that you need to know.

Watch this video to see an overview of horizontal discovery and top-down discovery.

Note: For this tutorial, the term discovery means horizontal discovery.

Probes, sensors, and patterns

Discovery uses these components to find CIs:

Probes and sensors

Basically, probes and sensors are scripts that collect and process data on a host and then update the CMDB. More specifically, probes explore or investigate CIs on your network, and sensors parse the data returned from the probes. Several probes and sensors are provided by default, but you can customize them to find different information, or you can create ones. You can also configure several parameters to control the behavior of a particular probe every time it is triggered.

Patterns

Patterns, like probes and sensors, are a series of operations that also collect data on a host, process it, and update the CMDB. Patterns differ from probes and sensors in that they are written in Neebula Discovery Language (NDL) rather than JavaScript, and they are called into action in
Horizontal discovery phases

The phases of horizontal discovery are:

Discovery follows these phases:

- **Scanning**
  Discovery sends a probed called Shazzam to the network to see if commonly used ports are open and if these ports can respond to queries. For example, if Shazzam finds a device that responds on port 135, Discovery knows that it is a Windows server.

- **Classification**
  If Discovery finds devices or computers, it sends additional probes to find the type of device or the operating system on the device. For example, Discovery sends the WMI probe to a Windows machine to detect the Windows 2012 operating system. Then Discovery uses records called classifiers, which specify the trigger probe or probes that run during the next two phases. If you are using patterns, the classifier specifies a trigger probe that in turn launches a pattern.

- **Identification**
  Discovery tries to gather more information about the device and then tries to determine if a CI for the device exists in the CMDB. Discovery then uses additional probes, sensors, and identifiers to update existing CIs in the CMDB or create new ones. Identifiers, also known as identification rules, specify the attributes that the probes look at when reconciling data with the CIs in the CMDB. If you are using patterns, Discovery uses the appropriate identification rule for the CI type specified in the pattern.

- **Exploration**
  The identifier launches additional probes configured in the classifier. These probes are especially designed as exploration probes to gather additional information about the device, like the applications running it, and additional attributes, such as memory, network cards, and drivers. Discovery then creates relationships between applications and devices and between applications. If you are using patterns, the operations in the pattern perform the exploration of the CI.

Discovery communication through MID Servers

A MID Server, which constantly queries the instance for probes to run, executes the instructions in the probe or in the pattern that the probe specifies. The MID Server then returns the results to the instance, where sensors process it. The MID Server does not retain any discovery information.

The MID Server starts all communications, using SOAP on HTTPS, which means that all communications are secure, and all communications are initiated inside the enterprise's firewall. No special firewall rules or VPNs are required.

Because Discovery is agentless, meaning that it does not require any permanent software to be installed on any computer or device to be discovered, the MID Server uses several techniques to probe devices without using agents. For example, the MID Server uses SSH to connect to a Unix or Linux computer, and then it can run a standard command, as specified in the probe, to gather information. Similarly, it uses the Simple Network Management Protocol (SNMP) to gather information from a network switch or a printer.
Types of discovery

The types of horizontal discovery that the Discovery application can perform are explained in the following table:

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network discovery</td>
<td>Run this type of discovery to find the internal IP networks within your organization. If you already know the IP address ranges in your network, it is not necessary to run network discovery.</td>
</tr>
<tr>
<td>CI discovery</td>
<td>Run this type of discovery to find the devices, computers, and applications on your network. This is essentially the standard type of discovery that you run most often.</td>
</tr>
<tr>
<td>Cloud discovery</td>
<td>Run this type of discovery to find AWS and Azure resources in your organization’s cloud.</td>
</tr>
<tr>
<td>Serverless discovery</td>
<td>Run this type of discovery to find applications on host machines without the need to discover the host first. Serverless discovery relies on patterns to explore CIs on a host.</td>
</tr>
</tbody>
</table>

IP service affinity

IP Service affinity saves the IP service information that is used to successfully find a device and associates it with the IP address of the device. Using this information, Discovery can target the device in subsequent runs with the accurate protocol. Discovery records the IP Service along with the IP address. Discovery can store the successful IP service information in the IP Service Affinity table (ip_service_affinity).

For example: A network device has both an SSH port and an SNMP port open. By its agentless design, Discovery tries SSH first. However, network devices should be discovered through SNMP. Discovery tries the SSH probe and it fails. This triggers the SNMP probe, which succeeds. With the association between the IP address and the IP service, subsequent discovery runs that target this IP address use SNMP first, because that is the probe that succeeded.

Help the help desk

Help the Help Desk is a standard feature available through the self-service Help the Help Desk application. It gathers information, much as Discovery does, about a single Windows computer by running a script on that computer. Discovery does many things that Help the Help Desk cannot do.
<table>
<thead>
<tr>
<th>Functionality</th>
<th>Discovery</th>
<th>Help the Help Desk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic discovery by schedule</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Automatic discovery on user login</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Manually initiated discovery</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Windows workstations</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Windows servers</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Linux systems</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Unix systems (Solaris, AIX, HP-UX, Mac (OSX))</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network devices (switches, routers, UPS, etc.)</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Printers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Automatic discovery of computers and devices</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Automatic discovery of relationships between processes running on servers</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

* When Discovery is installed.

**Horizontal discovery process flow with probes and sensors**

The horizontal discovery process passes through the four phases of discovery using probes, which gather information on the target machine, and then sensors, which help Discovery determine what to do with that information.

**Kicking off Discovery**

A user triggers horizontal discovery by configuring a discovery schedule or by launching an on-demand discovery with Discover now or Quick Discovery. The schedule specifies one or more IP addresses or range of IP addresses.

**Scanning phase**

1. Discovery first takes the Shazzam probe (and then port probes) and places it in a request in the External Communication Channel (ECC) queue.
2. The MID Server checks the ECC queue, retrieves the discovery request, and runs the probes against the host and discovers open ports.
3. The port probes scan common ports using several protocols, such as WMI, HTTP, SSH, and SNMP.
4. If one or more ports respond, the Shazzam probe sends information about the port back to the ECC queue through the MID Server.
5. Discovery checks the ECC queue to find out which ports responded, which identifies the type of machine. For example, if Shazzam detects that the machine is listening on port 22, Discovery treats the machine as a UNIX or Linux machine.

Classification phase

1. The Discovery application determines which classification probe to send to the newly discovered device by using information in the record of the port probe that successfully responded.
2. Discovery puts the classification probe into the ECC queue.
3. The MID Server checks the ECC queue, retrieves the discovery request, and runs the classification probe.
4. The classification probe retrieves additional information, such as which version of the operating system is running on a machine. This information determines the class of the CI that Discovery found. There is only one classification probe per discovered device.
5. The classification probe sends information back to the instance ECC queue through the MID Server.

The Identification phase

1. Discovery determines which classifier to use based on the class of the CI and the criteria specified in all CI classifier records. The classifier specifies which probes to use for the next two phases.
2. Discovery puts the identification trigger probe for the CI classifier into the ECC queue. For example, a Unix machine running HP-UX would require the HP-UX classifier, which specifies that the Multi Probe-HP-UX Identity identification trigger probe. These probes use identification rules to determine whether or not to insert or update a CI in the CMDB.

Note: The trigger probe could also be the Horizontal Pattern probe, which tells Discovery to follow the operations in the specified pattern, rather than sending out additional probes. The operations in the pattern cover both the identification and exploration phases. Discovery knows which identification rules to use based on the CI type, and Discovery makes inserts or updates to the CMDB based on these rules. Probes and sensors are not used.

3. The MID Server checks the ECC queue, retrieves the discovery request, and runs the identification trigger probe.
4. The identification probe accumulates identification data for each device and sends that data back to the instance via the MID Server.
5. Discovery uses sensors for the identifier probe to process the information.
6. Discovery performs the analysis on the CMDB using CI identifiers. Discovery can update existing CIs in the CMDB or create new ones.
The Exploration phase

1. Discovery looks at the Triggers Probes related list in the classifier to find exploration probes to run.
2. Discovery puts the exploration trigger probe into the ECC queue.
3. The MID Server checks the ECC queue, retrieves the discovery request, and runs the exploration trigger probes.
4. The probes send data back to the instance via the MID Server and sensors make updates to the CMDB, just as in the identification phase.

Horizontal discovery process flow with patterns

Horizontal discovery with patterns has four phases, just as horizontal discovery with probes does. However, for the last two phases, Discovery triggers operations from a pattern, rather than additional sets of probes.

Scanning phase

1. Discovery first takes the Shazzam probe (and then port probes) and places it in a request in the External Communication Channel (ECC) queue.
2. The MID Server checks the ECC queue, retrieves the discovery request, and runs the probes against the host and discovers open ports.
3. The port probes scan common ports using several protocols, such as WMI, HTTP, SSH, and SNMP.
4. If one or more ports respond, the Shazzam probe sends information about the port back to the ECC queue through the MID Server.
5. Discovery checks the ECC queue to find out which ports responded, which identifies the type of machine. For example, if Shazzam detects that the machine is listening on port 22, Discovery treats the machine as a UNIX or Linux machine.

Classification phase

1. The Discovery application determines which classification probe to send to the newly discovered device by using information in the record of the port probe that successfully responded.
2. Discovery puts the classification probe into the ECC queue.
3. The MID Server checks the ECC queue, retrieves the discovery request, and runs the classification probe.
4. The classification probe retrieves additional information, such as which version of the operating system is running on a machine. This information determines the class of the CI that Discovery found. There is only one classification probe per discovered device.
5. The classification probe sends information back to the instance ECC queue through the MID Server.
The Identification and Exploration phases

Patterns unify the Identification and Exploration phases of discovery.

1. Discovery determines which classifier to use based on the class of the CI and the criteria specified in all CI classifier records. The classifier specifies the Horizontal Pattern probe, which in turn specifies which pattern to launch. The Horizontal pattern probe also contains a sensor which does the actual work of updating the CMDB.

   **Note:** Patterns need applicative credentials to find applications running on host machines. Make sure you have applicative credentials configured along with the credentials required to access the host machine itself.

2. The operations in the pattern specify the actions that Discovery needs to take for both the identification and exploration phases. Discovery knows which identification rules to use based on the CI type in the pattern, and Discovery makes inserts or updates to the CMDB based on these rules. Only the Horizontal Discovery Sensor is used. Other probes and sensors are not used.

Devices and applications that Discovery can discover

In the base instance, Discovery can find and classify these devices and applications.

This list shows all device types and applications for Discovery only. For a detailed list of all supported applications for both Discovery and Service Mapping, including the protocol and the patterns or probes used for discovery, see Applications supported by Discovery and Service Mapping.

**Windows classifications**

- Windows Server 2008*
- Windows Server 2008 R2*
- Windows Server 2012*
- Windows Server 2012 R2*
- Windows Server 2016*
- Windows Server 2019
- Hyper-V Server*
- Windows Cluster Virtual IPs

**UNIX classifications**

- AIX*
- ESX
- HP-UX*
- Mac OS X
- Solaris*
- Linux*, including:
  - Red Hat
  - Fedora
  - Debian
  - SUSE
  - CentOS
Other OS classifications
z/OS*

Network device classifications
- DataPower servers*
- Dell Remote Access Controller
- Firewalls
- Netware servers
- Network printers
- Network router*
- Network switch*
- Power distribution units
- IP phones
- Wireless access points
- Uninterruptible Power Supplies (UPS)

Application classifications:

- **Web servers:**
  - Apache Tomcat, Apache mod_jk module, and Apache mod_proxy module*
  - Microsoft IIS*
  - Oracle (Sun) iPlanet*
  - JBoss*

- **Email and messaging services:**
  - IMAP
  - Exchange Client Access Server*, Exchange Hub*, Exchange MailBox*
  - Tibco Enterprise Message Service*

- **Cloud-based technology:**
  - Amazon Web Services*
  - Microsoft Azure*

- **Clusters:**
  - Oracle clusters*
  - Unix clusters*
  - Red Hat clusters*

- **Databases:**
  - MySQL*
  - DB2
  - Microsoft SQL
  - MongoDB
  - HBase
  - Oracle
• PostgreSQL*
• SAP HANA*
• Sybase*

• Storage servers:
  • SMI- Storage Server
  • SMI- Storage Switch
  • SMI- WBEM
  • NetApp Storage Server (7-mode and cluster mode)*

• Virtualization:
  • Docker*
  • Kernel-based Virtual Machine
  • Solaris Zones
  • vCenter

• Others:
  • Cisco Unified Computing System (UCS)*
  • Citrix License Server and Delivery Controller*
  • HP Service Manager application server*
  • HP Operations Manager*
  • Oracle JavaSpaces
  • GlassFish
  • Jrun*
  • LDAP service
  • MongoDB Shard (MongoS)*
  • NGINX
  • Puppet
  • SAP ASCS*, SAP Business Objects CMS Server*, SAP CI*, SAP DI*, SAP ERS*, SAP SCS*
  • Microsoft Sharepoint*
  • Microsoft SQL Server Analysis Services*
  • Oracle Tuxedo*
  • Oracle WebLogic*
  • IBM WebSphere*, WebSphere Message Broker*, IBM WebSphere MQ*
  • Tibco ActiveMatrix BusinessWorks*

Discovery uses patterns to find CIs noted with an asterisk *.

Applications supported by Discovery and Service Mapping
Discovery and Service Mapping can discover a wide range of operating systems and applications.

Discovery finds computers, servers, printers, a variety of IP-enabled devices, and the applications that run on them. It can then update the CIs in your CMDB with the data it collects. This discovery method is referred to as horizontal discovery. Service Mapping maps dependencies, based on a connection between devices and applications. This method is referred to as top-down mapping. The top-down mapping helps you immediately see the impact of a problematic object on the rest of the application service operation.

On top of hosts and applications supported by default, you can discover additional hosts and applications by deploying patterns available on Store.
If your organization uses devices or applications, which are not supported by default or using patterns available at Store, you can configure Discovery and Service Mapping to discover them as described in *Discovery patterns used by ITOM Visibility*.

<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>.Net Framework</td>
<td>Windows</td>
<td>1.x.x,2.x.x,3.x.x,4.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>.NET Application</td>
</tr>
<tr>
<td>Advanced Business Application Programming (ABAP) SAP Central Services (ASCS)</td>
<td>Windows, Linux, Hpux, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP ASCS Application patterns</td>
</tr>
<tr>
<td>Adobe JRun</td>
<td>Windows/Unix</td>
<td>4.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Jrun</td>
</tr>
<tr>
<td>Alteon RadWare ADC</td>
<td>Alteon</td>
<td>v31, v29.5</td>
<td>SNMP</td>
<td>Top-down and horizontal</td>
<td>Alteon Load Balancer</td>
</tr>
<tr>
<td>Amazon Relational Database Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Amazon AWS Relational Database Service</td>
</tr>
<tr>
<td>Amazon Route53 Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Horizontal</td>
<td>Amazon AWS Route53 HD</td>
</tr>
<tr>
<td>Amazon Application Load Balancer Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down</td>
<td>Amazon AWS application ELB service - TD</td>
</tr>
<tr>
<td>Amazon Elastic Load Balancer Service</td>
<td>AWS</td>
<td>N/A</td>
<td>REST</td>
<td>Horizontal</td>
<td>Amazon AWS classic ELB Service - TD</td>
</tr>
<tr>
<td>Apache HBase</td>
<td>Unix</td>
<td>N/A</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Probes:</td>
</tr>
<tr>
<td><em>Probes:</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><em>HBase - Version</em></td>
</tr>
<tr>
<td><em>Probes:</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><em>HBase - Main Class</em></td>
</tr>
<tr>
<td><em>Probes:</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><em>HBase - Configuration</em></td>
</tr>
<tr>
<td>Apache HTTP Server</td>
<td>Windows/Unix</td>
<td>2.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Apache</td>
</tr>
<tr>
<td>Apache Tomcat Servlet container HTTP web server</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Tomcat</td>
</tr>
<tr>
<td>Azure Database Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Azure Database</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>----------</td>
<td>---------</td>
<td>----------</td>
<td>----------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Azure DNS Zones Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Horizontal</td>
<td>Azure DNS Zones HD</td>
</tr>
<tr>
<td>Azure LoadBalancer Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Azure LoadBalancer TD</td>
</tr>
<tr>
<td>Azure WebSite Service</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down</td>
<td>Azure WebSite</td>
</tr>
<tr>
<td>BMC Control-M</td>
<td>Unix</td>
<td>6.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Cntl-M Enterprise Manager</td>
</tr>
<tr>
<td>CA eTrust Directory server</td>
<td>Windows</td>
<td>7.x, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>CA eTrust Directory server</td>
</tr>
<tr>
<td>CA Identity Manager Provisioning Server</td>
<td>Windows</td>
<td>11.x, 12.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>CA Identity Manager Provisioning Server for Windows</td>
</tr>
<tr>
<td>CA Introscope Enterprise Manager</td>
<td>Windows/Unix</td>
<td>8.x, 9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>CA Introscope Enterprise Manager</td>
</tr>
<tr>
<td>CA Policy Server</td>
<td>Windows/Unix</td>
<td>11.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>CA Policy Server</td>
</tr>
<tr>
<td>CA Site Minder Agent</td>
<td>Windows/Unix</td>
<td>11.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>CA Site Minder Agent</td>
</tr>
<tr>
<td>Cisco Content Services Switch Load balancer</td>
<td>ACE</td>
<td>6.x</td>
<td>SNMP/SSH</td>
<td>Top-down and horizontal</td>
<td>Cisco CSS SNMP</td>
</tr>
<tr>
<td>Cisco ACE Application Control Engine</td>
<td>ACE</td>
<td>2.x</td>
<td>SSH</td>
<td>Top-down and horizontal</td>
<td>ACE Load Balancer</td>
</tr>
<tr>
<td>Cisco Call Manager</td>
<td>Cisco</td>
<td>3.x, 4.x</td>
<td>SNMP/SSH</td>
<td>Top-down</td>
<td>Cisco CallManager</td>
</tr>
<tr>
<td>Cisco Fiber Connect</td>
<td>Cisco</td>
<td>No data</td>
<td>SNMP/SSH</td>
<td>Top-down</td>
<td>Cisco Fibre Connect</td>
</tr>
<tr>
<td>Cisco Global Site Selector Load Balancer</td>
<td>ACE</td>
<td>3.x</td>
<td>SNMP/SSH</td>
<td>Top-down and horizontal</td>
<td>Cisco GSS</td>
</tr>
<tr>
<td>Cisco Unified Computing System</td>
<td>UCS</td>
<td>3.x</td>
<td>REST</td>
<td>Horizontal</td>
<td>UCS - HD</td>
</tr>
<tr>
<td>Citrix Delivery Controller</td>
<td>Windows</td>
<td>7.x&gt;7, 5, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Delivery Controller</td>
</tr>
<tr>
<td>Citrix Usage Collector</td>
<td>Windows</td>
<td>7.x&gt;7, 5, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Citrix Collector</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------</td>
<td>--------------------------</td>
<td>-------------------</td>
<td>---------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Citrix Netscaler Global Server Load Balancing (GSLB)</td>
<td>Netscaler</td>
<td>10.x</td>
<td>SNMP/SSH/NS.conf</td>
<td>Top-down and horizontal</td>
<td>Netscaler GLB</td>
</tr>
<tr>
<td>Citrix License Server</td>
<td>Windows</td>
<td>7.x</td>
<td>WMI</td>
<td>Horizontal</td>
<td>License Server</td>
</tr>
<tr>
<td>Citrix Netscaler Load Balancer</td>
<td>Netscaler</td>
<td>9.x, 10.x</td>
<td>SNMP/SSH/NS.conf</td>
<td>Top-down and horizontal</td>
<td>Citrix Netscaler</td>
</tr>
<tr>
<td>Citrix Presentation Server</td>
<td>Windows</td>
<td>4.x, 5.x, 6.x&lt;6.5</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Citrix XenApp or Presentation Server</td>
</tr>
<tr>
<td>Citrix XenApp Server</td>
<td>Windows</td>
<td>6.x&gt;6.5, 7.x&lt;7.6</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Citrix XenApp or Presentation Server</td>
</tr>
<tr>
<td>Connect APK</td>
<td>Windows</td>
<td>1.x.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Connect-It Service</td>
</tr>
<tr>
<td>Docker Container Linux</td>
<td>Docker</td>
<td>No data</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Docker Pattern</td>
</tr>
<tr>
<td>Epic Systems Corporation</td>
<td>Unix</td>
<td>2014.x.x, 2015.x.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>EPIC Cache</td>
</tr>
<tr>
<td>F5 BIG-IP Device Service Clustering</td>
<td>F5</td>
<td>11.x, 12.x</td>
<td>SNMP/SSH/REST</td>
<td>Horizontal</td>
<td>F5 Cluster</td>
</tr>
<tr>
<td>F5 BIG-IP Load balancer</td>
<td>F5</td>
<td>11.x, 12.x</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>F5 Load Balancer</td>
</tr>
<tr>
<td>HAProxy Server</td>
<td>Unix</td>
<td>1.5, 1.6, 1.7</td>
<td>SSH</td>
<td>Top-down</td>
<td>HA Proxy</td>
</tr>
<tr>
<td>HP Operations Manager</td>
<td>Windows/Unix</td>
<td>9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>HP Operations Manager</td>
</tr>
<tr>
<td>HP Quality Center</td>
<td>Windows</td>
<td>10.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>HP Quality Center on Windows</td>
</tr>
<tr>
<td>HP Service Manager software</td>
<td>Windows/Unix</td>
<td>9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>HP Service Manager Application Server</td>
</tr>
<tr>
<td>HP UCMDB</td>
<td>Windows</td>
<td>No data</td>
<td>WMI/WinRM</td>
<td></td>
<td>HP uCMDB on Windows</td>
</tr>
<tr>
<td>HP-UX Unix Server</td>
<td>HPUX</td>
<td>11i</td>
<td>SSH</td>
<td>Horizontal</td>
<td>HP-UX Server</td>
</tr>
<tr>
<td>IBM AIX</td>
<td>Unix</td>
<td>6.x, 7.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>AIX Server</td>
</tr>
<tr>
<td>IBM CICS Transaction Gateway</td>
<td>Windows/Unix</td>
<td>5.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>CTG</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------</td>
<td>------------------</td>
<td>------------------</td>
<td>-------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>IBM Customer Information Control System</td>
<td>Windows/Unix</td>
<td>4.x, 5.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>CICS</td>
</tr>
<tr>
<td>IBM DB2</td>
<td>Windows/Unix</td>
<td>8.x, 9.x, 10.x, 11.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>DB2</td>
</tr>
<tr>
<td>IBM Security Access Manager</td>
<td>ISAM</td>
<td>8.0.1</td>
<td>SNMP/REST</td>
<td>Top-down and horizontal</td>
<td>ISAM Server</td>
</tr>
<tr>
<td>IBM Tivoli Access Manager</td>
<td>Windows/Unix</td>
<td>6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>ITAM - Asset Center</td>
</tr>
<tr>
<td>IBM Tivoli WebSEAL</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Webseal</td>
</tr>
<tr>
<td>IBM Websphere Application server</td>
<td>Windows/Unix</td>
<td>5.x, 6.x, 7.x, 8.x, 9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Websphere</td>
</tr>
<tr>
<td>IBM Websphere DataPower SOA Appliance</td>
<td>DataPower</td>
<td>140, 150</td>
<td>SNMP/REST</td>
<td>Top-down and horizontal</td>
<td>DataPower</td>
</tr>
<tr>
<td>IBM Websphere Message Queue</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x, 9.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>WNQ</td>
</tr>
<tr>
<td>IBM Websphere Portal</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Websphere Portal On Windows</td>
</tr>
<tr>
<td>IBM Z mainframe</td>
<td>Z/OS</td>
<td>OS/390</td>
<td>SSH</td>
<td>Horizontal</td>
<td>IBM zOS Server</td>
</tr>
<tr>
<td>Interconnect Web Service</td>
<td>Windows</td>
<td>2014</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Inter Connect</td>
</tr>
<tr>
<td>JBoss Application Server</td>
<td>Windows/Unix</td>
<td>6.x, 7.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Jboss pattern</td>
</tr>
<tr>
<td>Linux Server</td>
<td>Unix</td>
<td>Ubuntu 16, RHEL 5.x, CentOS 6.x, 7.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Linux Server</td>
</tr>
<tr>
<td>Microsoft Active Directory Domain Services</td>
<td>Windows</td>
<td>13, 30, 31, 44, 47, 52, 56, 69</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Active Directory Service pattern</td>
</tr>
<tr>
<td>Microsoft Azure SQL Database</td>
<td>Azure</td>
<td>N/A</td>
<td>REST</td>
<td>Top-down and horizontal</td>
<td>Cloud Database Instance</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------------</td>
<td>-----------------------------</td>
<td>--------------</td>
<td>----------------------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>Microsoft Fast Search</td>
<td>Windows</td>
<td>2005</td>
<td>WMI/WinRM</td>
<td>Horizontal and top-down</td>
<td>Fast Search</td>
</tr>
<tr>
<td>Microsoft Identity Integration Server</td>
<td>Windows</td>
<td>2003</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>IIFP On Windows pattern</td>
</tr>
<tr>
<td>Microsoft Internet Information Services</td>
<td>Windows</td>
<td>6.0, 7.x, 8.x</td>
<td>WMI/WinRM</td>
<td>Top-down and horizontal</td>
<td>IIS</td>
</tr>
<tr>
<td>Microsoft Message Queueing</td>
<td>Windows</td>
<td>3.x, 4.x, 5.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>MSMQ pattern</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MongoDB</td>
<td>Windows/Unix</td>
<td>3.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>mongos server</td>
</tr>
<tr>
<td>Netapp Storage</td>
<td>Netapp</td>
<td>7.x, 8.x</td>
<td>SNMP/Basic Auth</td>
<td>Horizontal</td>
<td>NetApp Storage 7-Mode</td>
</tr>
<tr>
<td>Nginx Web server, proxy server</td>
<td>Windows/Unix</td>
<td>1.10, 1.10.1</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Nginx Pattern</td>
</tr>
<tr>
<td>OpenText Documentum</td>
<td>Windows/Unix</td>
<td>6.x, 7.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Pattern for license server</td>
</tr>
<tr>
<td>Oracle Application Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Oracle iAS</td>
</tr>
<tr>
<td>Oracle Concurrent Server</td>
<td>Windows/Unix</td>
<td>10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Concurrent Server</td>
</tr>
<tr>
<td>Oracle Clusterware</td>
<td>Linux</td>
<td>10.x, 11.x, 12.x</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Oracle CRS</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>Windows/Unix</td>
<td>8.x.x, 9.x.x, 10.x.x, 11.x.x, 12.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Oracle DB</td>
</tr>
<tr>
<td>Oracle Database Advanced Queueing</td>
<td>Windows/Unix</td>
<td>9.x.x, 10.x.x, 11.x.x, 12.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Advanced Queue Queue</td>
</tr>
<tr>
<td>Oracle Discover Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Discoverer Engine</td>
</tr>
<tr>
<td>Oracle Forms</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Forms Engine</td>
</tr>
<tr>
<td>Oracle Fulfillment Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Fulfillment Server</td>
</tr>
<tr>
<td>Oracle GlassFish Server</td>
<td>Windows/Unix</td>
<td>3.1, 4.0, 4.1</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>GlassFish Server</td>
</tr>
<tr>
<td>Oracle HTTP Server</td>
<td>Windows/Unix</td>
<td>No data</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>HTTP Server</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------</td>
<td>---------------------</td>
<td>-------------------</td>
<td>-------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Oracle iPlanet Web Server</td>
<td>Solaris</td>
<td>4.x, 6.x, 7.x</td>
<td>SSH</td>
<td>Top-down and horizontal</td>
<td>Sun iPlanet Webserver</td>
</tr>
<tr>
<td>Oracle Java Enterprise System(JES)</td>
<td>Solaris</td>
<td>7.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>Sun JES pattern</td>
</tr>
<tr>
<td>Oracle Metric Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Metric Server</td>
</tr>
<tr>
<td>Oracle MySQL Server</td>
<td>Windows/Unix</td>
<td>4.x, 5.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>My SQL server On Windows and Linux Pattern</td>
</tr>
<tr>
<td>Oracle Notification Server</td>
<td>Windows/Unix</td>
<td>9.x, 10.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Oracle Notification Server</td>
</tr>
<tr>
<td>Oracle Peoplesoft</td>
<td>Windows/Unix</td>
<td>8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Peoplesoft Application Server</td>
</tr>
<tr>
<td>Oracle Process Manager</td>
<td>Windows/Unix</td>
<td>9.x, 10.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Process Manager</td>
</tr>
<tr>
<td>Oracle RAC</td>
<td>Windows/Unix</td>
<td>9.x, 10.x.x, 11.x.x, 12.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>App TNS Service</td>
</tr>
<tr>
<td>Oracle Report Server</td>
<td>Windows/Unix</td>
<td>9.x.x, 10.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Report Server</td>
</tr>
<tr>
<td>Oracle Solaris Server</td>
<td>Solaris</td>
<td>9, 10, 11, SPARC</td>
<td>SSH</td>
<td>Horizontal</td>
<td>Solaris Server</td>
</tr>
<tr>
<td>Oracle Tuxedo</td>
<td>Windows/Unix</td>
<td>10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Tuxedo</td>
</tr>
<tr>
<td>Oracle Tuxedo Portal</td>
<td>Windows/Unix</td>
<td>10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>Tuxedo Portal pattern</td>
</tr>
<tr>
<td>Oracle Weblogic Application Server</td>
<td>Windows/Unix</td>
<td>8.x, 9.x, 10.x, 11.x, 12.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>WebLogic</td>
</tr>
<tr>
<td>Pivotal RabbitMQ</td>
<td>Windows/Unix</td>
<td>3.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down</td>
<td>RabbitMQ Cluster Unix pattern</td>
</tr>
<tr>
<td>PostgreSQL DB</td>
<td>Windows/Unix</td>
<td>6.x, 7.x, 8.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>PostgreSQL DB</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------</td>
<td>---------</td>
<td>----------</td>
<td>----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Puppet                  | Unix             | N/A     | SSH      | Horizontal     | - Puppet – Master Info  
- Puppet – Certificate Requests  
- Puppet – Manifest  
- Puppet – Modules                                      |
<p>| RadWare AppDirector     | RadWare          | No data | SNMP     | Top-down and horizontal | AppDirector Load Balancer |
| Load Balancer           |                  |         |          |                |                                                                                                               |
| Red Hat Cluster Suite   | Linux            | RH 5.x, 6.x, 7.x | SSH      | Horizontal     | RH Cluster                                                                                                   |
| SAP Convergent Invoicing (CI), including: | Windows, Linux, Hpx, Solaris, AIX | 7.x.x | WMI/WinRM/SSH | Horizontal and top-down | SAP CI Application pattern |
|                         |                  |         |          |                |                                                                                                               |
|                         |                  |         |          |                |                                                                                                               |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Platform</th>
<th>Version</th>
<th>Protocol</th>
<th>Discovery type</th>
<th>Pattern (or probe if indicated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP NetWeaver Development Infrastructure (DI), including:</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP DI Application pattern</td>
</tr>
<tr>
<td>Change Management Service (CMS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design Time Repository (DRT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component Build Service (CBS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Landscape Directory (SLD and Name Service)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Common user management of the development infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integration of the transports with the Change Transport System (CTS) of Application Server ABAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Available tools for monitoring the development infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAP Java Connector (JC)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP JC Application pattern</td>
</tr>
<tr>
<td>SAP Evaluated Receipt Settlement (ERS)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP ERS Application pattern</td>
</tr>
<tr>
<td>SAP Central Services (SCS)</td>
<td>Windows, Linux, Hpx, Solaris, AIX</td>
<td>6.x.x, 7.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Horizontal and top-down</td>
<td>SAP SCS Application pattern</td>
</tr>
<tr>
<td>Name</td>
<td>Platform</td>
<td>Version</td>
<td>Protocol</td>
<td>Discovery type</td>
<td>Pattern (or probe if indicated)</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------</td>
<td>------------------</td>
<td>----------------</td>
<td>-------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>SAP Sybase DB</td>
<td>Windows/Unix</td>
<td>4.x, 5.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Sybase</td>
</tr>
<tr>
<td>Sun Directory Proxy Server</td>
<td>Solaris</td>
<td>No data</td>
<td>SSH</td>
<td>Top-down</td>
<td>Sun Directory Proxy Server</td>
</tr>
<tr>
<td>Sun Java System Directory Server</td>
<td>Solaris</td>
<td>No data</td>
<td>SSH</td>
<td>Top-down</td>
<td>Sun Directory Server</td>
</tr>
<tr>
<td>Thunder ADC</td>
<td>A10</td>
<td>2.x, 4.x</td>
<td>SNMP/SSH</td>
<td>Top-down and horizontal</td>
<td>A10 Load Balancer</td>
</tr>
<tr>
<td>TIBCO ActiveMatrix Adapter</td>
<td>Unix</td>
<td>7.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>Tibco Adapter</td>
</tr>
<tr>
<td>TIBCO ActiveMatrix BusinessWorks</td>
<td>Windows/Unix</td>
<td>5.x.x, 6.x.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>ActiveMatrix Business Works</td>
</tr>
<tr>
<td>TIBCO Enterprise Message Service (EMS)</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Enterprise Message Service</td>
</tr>
<tr>
<td>TIBCO Enterprise Message Service (EMS) Queue</td>
<td>Windows/Unix</td>
<td>5.x, 6.x</td>
<td>WMI/WinRM/SSH</td>
<td>Top-down and horizontal</td>
<td>Enterprise Message Service</td>
</tr>
<tr>
<td>TIBCO Hawk</td>
<td>Unix</td>
<td>4.x, 5.x</td>
<td>SSH</td>
<td>Top-down</td>
<td>TibcoHawkPattern</td>
</tr>
<tr>
<td>Veritas Enterprise Vault</td>
<td>Windows</td>
<td>7.x</td>
<td>WMI/WinRM</td>
<td>Top-down</td>
<td>Enterprise Vault</td>
</tr>
</tbody>
</table>

**Request Discovery**

Discovery is available as a separate subscription from the rest of the Now Platform and requires the Discovery (com.snc.discovery) plugin.

Role required: admin
To purchase a Discovery subscription, contact your ServiceNow account manager or sales representative. When the Discovery plugin is activated, the following plugins are also activated:

- The Cloud Management Core (com.snc.cloud.core) plugin is also activated to provide functionality for Azure and AWS discovery
- The ITOM Licensing (com.snc.itom.license) plugin

1. From your instance, navigate to System Definition > Plugins.
2. On the All Applications page, click Request Plugin to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>Date and time must be at least two business days from the current time.</td>
</tr>
<tr>
<td>Reason/Comments</td>
<td>Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments.</td>
</tr>
</tbody>
</table>

5. Click Submit.

Pattern updates for the New York release

Several new patterns are available for horizontal discovery.

Patterns for newly discovered CIs

These patterns are new for the New York release:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UNIX Cluster - ORACLE Clusterware</strong></td>
<td>This pattern finds Oracle cluster software that runs Real Application Clusters (RAC) for Oracle databases. This pattern supports the discovery of High-availability Service Management that includes the cluster, nodes, resource information, and status.</td>
</tr>
<tr>
<td><strong>UNIX Cluster - Linux Red hat cluster and Linux - Red Hat Cluster</strong></td>
<td>These patterns find Linux Red Hat clusters that offer high availability and load balancing.</td>
</tr>
<tr>
<td><strong>F5 LB REST</strong></td>
<td>This pattern finds F5 BigIP load balancers via HTTP/HTTP classification, which is prioritized after SNMP and SSH. This pattern requires a GET request from the HTTP - Classify port probe.</td>
</tr>
</tbody>
</table>
Pattern changes

These patterns were modified in this release:

<table>
<thead>
<tr>
<th>Pattern name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Netscaler Load Balancer</td>
<td>Populates DNS names and aliases.</td>
</tr>
<tr>
<td>F5 Load Balancer</td>
<td>Includes a new section to find F5 clusters via SNMP.</td>
</tr>
<tr>
<td>Router and switch discovery</td>
<td>Added support for router stacks. Discovery creates a Master of::Stack Member of relationship between two switches.</td>
</tr>
</tbody>
</table>

Discovery setup

After you activate the Discovery application, you have several ways to get started.

Select your method of setup:

<table>
<thead>
<tr>
<th>Setup method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guided Setup</td>
<td>Use this method if you want to set up everything you need to use Discovery, including a MID Server. Discovery guided setup provides a sequence of tasks that help you configure Discovery on your ServiceNow instance. To open Discovery guided setup, navigate to Guided Setup &gt; ITOM Guided Setup. For more information about using the guided setup interface, see Using guided setup.</td>
</tr>
<tr>
<td>Without Guided Setup</td>
<td>Use this method if you want to set up each component of setup separately. You can configure MID Servers, credentials, and roles separately, then configure one of the discovery types from the Discovery Schedule.</td>
</tr>
<tr>
<td>Discovery Quick Start</td>
<td>Use this wizard to configure everything you need to perform subnet and CI Discovery in your environment. Before you can use Quick Start, you must first install and validate your MID Servers and create the necessary credentials.</td>
</tr>
</tbody>
</table>

Set up Discovery without Guided Setup

If you are not using Guided Setup, you must perform several configuration steps manually to activate the application, set up the MID Server, and then set up Discovery.
Grant Discovery role: Verify that users who are expected to configure and execute Discovery in your network have the discovery_admin role. This role grants access to the tables in the Discovery application.

Designate a computer to host the MID Server.

Gather the login credentials that the MID server must use to access the devices it is expected to discover.

Determine which IP addresses Discovery must scan. In a highly complex network, group IP ranges into a range sets, which are reusable for different schedules. Discovery does not scan anything outside of these ranges.

Role required: discovery_admin

1. Install one or more MID Servers on physical or virtual servers that meet the minimum requirements and configure them to communicate with the appropriate ServiceNow instance.
   Install at least two MID Servers at first and assign them to different schedules and IP ranges to help complete discoveries quicker. Starting with the Istanbul release, you must make sure that the MID Server is assigned the Discovery application or the ALL application.

2. Ensure MID Server connectivity by navigating to Discovery > MID Servers.
   If the MID Servers are configured properly, they appear in the list of MID Servers.

3. If a MID Server does not appear as a choice in the instance, perform the following checks:
   - Ensure that the URL in the Configuration Parameters related list provided is correct.
   - If the MID Server is installed on Windows, make sure the ServiceNow MID Server service is configured properly with the correct logon credentials and is running.
   - Check the MID Server log for errors.
     - On the instance, navigate to Discovery > MID Servers > (MID Server) > Logs.
     - On the MID Server host, navigate to the agent\logs\agent0.log0 directory.
   - If Basic Authentication is enabled, a user name and password must be provided.
   - The MID Server might not have outbound access on port 443 (SSL) or a proxy server might be preventing TCP communication to the instance.
   - Make sure that no firewalls are blocking communication between the MID Server and the instance.

4. Set the credentials on the instance for all the computers, printers, and network gear in your environment.
   Credentials for Windows devices (using the WMI protocol) are provided by the logon configured for the MID Server service on the Windows server host. Credentials for UNIX, vCenter, and SNMP must be configured on the instance. Discovery automatically determines which credentials work for a particular computer or device.

5. Define and run Discovery schedules.
   The Discovery Schedule is the control point for running discoveries. The schedule controls when Discovery runs, defines the MID Server to use, the type of Discovery that should run, and the IP addresses to query. Create as many schedules as necessary, using different types of discoveries, and configure them to run at any time. Let Discovery run on its configured schedule or manually execute Discovery at any time. You can set up a single schedule for the entire network or separate schedules for each location or VLAN. If you do not know the IP address to scan in your network, run a Network Discovery first to discover the IP networks. Once discovered, you can convert these networks into IP address range sets that you use in a Discovery Schedule.
6. Validate the Discovery results.
   Open the ECC queue and analyze the XML payload, then check the Discovery log. For more information, refer to the section for validating results in Schedule a horizontal discovery.

Discovery Quick Start

Discovery Quick Start is a wizard that helps you get up and running with Discovery quickly. Use Discovery Quick Start tasks to discover IP ranges, or subnets, in your environment and then to identify the configuration items (CI) in those subnets.

Before using Discovery Quick Start, install and validate MID Servers. See MID Server Installation for details.

Configure and schedule subnet discovery
Subnet discovery finds all the subnets in your environment to determine available IP ranges and to ensure that there are MID Servers available to access these ranges.

Role required: discovery_admin

The goal of this phase is to discover all the subnets in your network. To do this, Discovery must have the correct SNMP credentials to identify the routers it encounters. Using the credentials you provide, Discovery begins by reading the router tables of the router connected to the MID Server to identify connected routers. Discovery then moves outward through the network from router to router until it has discovered all the local subnets.

After Discovery identifies the subnets, it automatically assigns the IP ranges of these subnets to qualified MID Servers in preparation for the CI discovery phase.

1. Navigate to Discovery > Home.
   The first time you access Discovery Home, a welcome screen appears. After you have configured subnet discovery and run the schedule, Quick Start shows the subnet results information.

2. Click Discovery Quick Start to begin configuration.
3. In the **Select MID Servers** tab, select qualified MID Servers for subnet discovery.
   a) Move the MID Servers you want to use for subnet discovery from the **Available** column to the **Selected** column.

   Available MID Servers must meet these requirements:
   
   - Be configured and in an **Up** state.
   - Be validated.
   - Have the Discovery application attached or **ALL** applications configured.
b) If you want Discovery to use all new MID Servers that meet the setup criteria, select the **Automatically include new MID Servers** option.

c) Click **Next**.

4. In the **Add Credentials** tab, add credentials for the routers you are going to discover. This tab shows any SNMP credentials that are already configured. Routers that appear in the **Targeted Routers** list are default gateway routers to which the MID Server host is connected.

**Note:** You can return to this phase after creating a schedule to add credentials.
a) Click **New** to add SNMP community or SNMP V3 credentials, if necessary.

b) If the list of targeted routers does not include all the routers you expect to use, click the plus icon and add the router by IP address.
c) After you configure all necessary credentials, click **Test Router Access** to verify that Discovery can use your credentials to access routers.

The gateway routers appear in the **Routers** column, regardless of whether the credentials can access the router. Those routers that the credentials can successfully access appear with a **Success** indicator. The routers that cannot be reached, appear with a **Failed** indicator.

You can filter the list using the **Search by IP** field or by using the **Reached Routers** or **Unreached Routers** options from the **View** choice list.

d) Click **Next**.

5. In the **Create Schedule** tab, create a schedule to run subnet discovery regularly.

Execution of this schedule performs subnet discovery and triggers IP subnet auto-assignment, in which the discovered subnets are assigned to qualified MID Servers.

a) Complete the schedule using the fields from the list.
• **Active**: Enables or disables this schedule.

• **Frequency**: Select how often the schedule should run:
  
  • **Daily**: Run Discovery at a specified time each day.
  
  • **Weekly**: Run Discovery on one designated day of each week.
  
  • **Monthly**: Run Discovery on one designated day of the month. If the designated day does not occur in the month, the schedule does not run that month. For example, if you designate day 30, the schedule does not run in February.
  
  • **Once**: Run Discovery once on the designated date and time.
  
  • **Periodically**: Run Discovery every designated period of time. Use the **Repeat Interval** field to define the period of time in days, hours, minutes and seconds. The first discovery runs at the point in time defined in the **Starting** field. The subsequent discoveries run after each **Repeat Interval** period passes.

• **Cancel Discovery if longer than**: Terminate the Discovery process if it exceeds the specified duration. Discovery might run for an excessive amount of time based on multiple factors, including the size of your network.

  **Note**: Any MID Server processing taking place when Discovery is canceled is allowed to complete. However, all output records that MID Servers have not picked up are canceled.

b) Click one of these:

  • **Finish and Run** saves the schedule and runs discovery immediately.
  
  • **Finish** saves the schedule. Discovery then runs at the interval specified in the schedule.

6. View subnet discovery results.
a) Click each tile for details. If subnet Discovery did not yet run, each of these tiles shows 0.

- **Reached Routers**: The number of routers that Discovery was able to access and discover.
- **Unreached Routers**: The number of routers that Discovery was not able to access.
- **Found Subnets**: The number of subnets that Discovery found.
- **Assigned Subnets**: IP subnets that were assigned to at least one MID Server.
- **Unassignable Subnets**: IP subnets that were not assigned to any MID Servers.

b) Click **Edit Schedule** to return to the **Select MID Servers** tab.

c) Resolve any issues with subnet discovery by returning to the Router Access page and making sure each credential is configured correctly. Also test router access for each credential.

d) Click **Advanced View** to see the Automation Status Set form. This form displays detailed results for both subnet discovery and the automatic assignment of IP ranges.

Configure and schedule CI discovery

Using the credentials you provide, the Quick Start wizard finds all the configuration items (CI) for the IP ranges identified during subnet discovery.

Role required: discovery_admin

You should perform this procedure after you complete Configure and schedule subnet discovery in the Discovery Quick Start.

The goal of this phase is to create the schedules necessary to discover the CIs on each subnet. Discovery Quick Start does that automatically, and then runs these schedules using the credentials you provide. You can add additional credentials at any time. Optionally, you can
define locations for each of the subnet routers, and Discovery creates separate schedules for each location.

You must be familiar with the types of credentials that you need for the CIs in your network. Each CI type requires different credentials. For example, if Discovery uses patterns instead of probes, the patterns might require **applicative credentials**.

Review the requirements for the CI types you intend to discover. See *Data collected by Discovery*, and then locate the specific type of device, computer, or application.

1. In the **Add Credentials** tab in the Discovery Quick Start, create the credentials necessary to discover CIs in your network.
   a) Select the credential type you want to manage from the **Type** field.

   All the credentials currently configured for that CI type appear in rows of tiles.
   
   b) To add a credential, click **New**.

   The Add Credential form appears, allowing you to specify your new credential.

   c) Click **Next** to go to the **Router Location** phase.
2. **Optional:** In the **Assign Router Location** tab, assign a location to each router that subnet discovery found.

When you configure router locations, you have the option of creating a schedule for each location. If you do not assign locations, a single Discovery schedule is created for all your subnets. If you assign a timezone to a location, Discovery creates a schedule that runs in that location’s timezone. If you do not assign a timezone to a location, the schedule for that location runs in the user’s local timezone.

**Note:** A subnet can belong to multiple routers. If schedules are created by location, Discovery finds subnets managed by multiple routers using multiple schedules.

The list of gateway routers shows those devices not currently assigned to a location.

![Image of Assign Routers to Locations](image-url)

<table>
<thead>
<tr>
<th>Unassigned Routers (4/4)</th>
<th>for next</th>
<th>search</th>
<th>Select All</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td><strong>Manufacturer</strong></td>
<td><strong>IP Address</strong></td>
<td><strong>Serial Number</strong></td>
</tr>
<tr>
<td>IBM-001</td>
<td>IBM</td>
<td>10.218.64.3</td>
<td>6.7.80.1</td>
</tr>
<tr>
<td>IBM-002</td>
<td>IBM</td>
<td>10.218.241.33</td>
<td>6.7.80.1</td>
</tr>
<tr>
<td>IBM-003</td>
<td>IBM</td>
<td>10.218.64.3</td>
<td>6.7.80.1</td>
</tr>
<tr>
<td>IBM-004</td>
<td>IBM</td>
<td>10.218.241.33</td>
<td>6.7.80.1</td>
</tr>
</tbody>
</table>

- a) Select the check boxes for the routers you want to assign a location.
  
  Use the **System Location** column in the table to help you assign a location to the router.

- b) Select the location in the **Assign to** choice list.

- c) If the location is not already in the list, click **Add New Location** and add the location.

- d) Ensure that the location has a timezone defined.
  
  Timezones allow you to create a schedule for each location based on the timezone.

- e) Click **Assign**.
f) Click **Next**.

3. Create a CI Discovery schedule.
   a) Complete the schedule using the fields in the list.

- **Active**: Enables or disables this schedule.
- **Frequency**: Select how often the schedule should run:
  - **Daily**: Run Discovery at a specified time each day.
  - **Weekly**: Run Discovery on one designated day of each week.
  - **Monthly**: Run Discovery on one designated day of the month. If the designated day does not occur in the month, the schedule does not run that month. For example, if you designate day 30, the schedule does not run in February.
  - **Once**: Run Discovery once on the designated date and time.
  - **Periodically**: Run Discovery every designated period of time. Use the **Repeat Interval** field to define the period of time in days, hours, minutes and seconds. The first discovery runs at the point in time defined in the **Starting** field. The subsequent discoveries run after each **Repeat Interval** period passes.

- **Create a separate schedule for each location**: Automatically create schedules for any router locations you have configured. These schedules run on the day and time...
you specify, but in their own local timezone. Clear this check box to create a single schedule for all your locations that runs in the user's local timezone. This option is only available if you have configured router locations.

- **Cancel Discovery if longer than**: Terminate the Discovery process if it exceeds the specified duration. Discovery might run for an excessive amount of time based on multiple factors, including the size of your network.

  Note: Any MID Server processing taking place when Discovery is canceled is allowed to complete. However, all output records that MID Servers have not picked up are canceled.

b) Click one of these options to finish the Quick Start:

- **Finish and Run** saves the schedule and runs a discovery immediately.
- **Finish** saves the schedule. Discovery runs at the interval you specified in the schedule.

### Running discoveries in your network

You can run discoveries from schedules or scripts to create configuration items, define subnets, or to find resources in AWS and Azure clouds.

### MID Server configuration prerequisites

Ensure that your MID Servers are properly configured prior to creating a Discovery schedule.

- **Supported applications**: Select the applications that are allowed to use the MID Server. You can use the **ALL** application option to allow any application to use the MID Server.
- **IP ranges**: Define the ranges of IP address the MID Server can scan. To find a MID Server match, the IP range you configure on the Discovery schedule must fall into the ranges that one or more MID Servers can support.
- **Capabilities**: Create the capabilities that the MID Server supports. You can use the **ALL** capability option to allow any application to use the MID Server.

### Discovery configuration prerequisites

Ensure that your MID Servers can authenticate on the devices they find and classify configuration items (CI) properly.

- **Credentials**: Configure the MID Servers with the login credentials they need to query the devices in the network.
- **Classifications**: The device and process classifications provided in the base platform are normally sufficient. Create classifications as needed for the devices, processes, and applications in the network not covered by the default classifiers.

### Get started running a discovery

1. Use the Discovery Configuration Console to get started with Discovery. The console provides configuration options which let you choose the types of devices, applications, software files, and software CIs you want Discovery to find. If you select a CI to exclude from scanning, the
instance disables the related probe or classifier that Discovery uses to identify the CI. See *Discovery Configuration Console* to get started.

2. Determine what type of discovery to run:

- **Run a Configuration item (CI) discovery** to find the devices, computers, and applications on your network. This is the most common type of discovery. Run CI discovery from the Discovery Schedule, where you set up a recurring schedule or run a discovery on demand. The Discovery Schedule also provides configuration options for MID Servers and the Shazzam port probe.

- **Run a Network Discovery** to find the internal IP networks within your organization. If you already know the IP address ranges in your network, it is not necessary to run Network Discovery. It is intended for organizations that do not have complete knowledge of the IP addresses available for Discovery in their networks.

- **Run a Serverless Discovery** to find applications on host machines without the need to discover the host first. Serverless Discovery relies on infrastructure patterns to explore CIs on a host. This kind of discovery skips the scanning and classification phases of discovery. You need an advanced knowledge of patterns to use this type of discovery. Refer to *Patterns and horizontal discovery* to get started with patterns.

3. After you run a discovery, monitor the results of the discovery and resolve errors if they occurred:

- **Use the Discovery status** to see a summary of a Discovery and to access the ECC queue, which shows probe and sensor activity, as well as the actual XML payload that is sent to or from an instance.

- **Use the Discovery dashboard** to monitor ongoing Discovery operations.

- Use the **Discovery Home page** to access details for all schedules, cloud resources (virtual machines), discovered devices, and related errors that might have occurred. Error details include possible remediation steps.

**Discovery Configuration Console**

Use the Discovery Configuration Console to manage what kind of CIs and CI information you want to discover.

By default, Discovery finds all the information on your network that is specified in probes and patterns. Use the controls in this console to select the data Discovery adds to the CMDB that your organization needs. You can control these aspects of Discovery:

- The discovery of entire CI types, such as Windows servers.
- The discovery of specific CI details, such as OS information on Windows servers.
- The discovery of software packages containing keyword terms that you add, such as Hotfix or Security Update.
- File-based Discovery of software on UNIX and Windows by particular extension or folder, such as .cert or .ext.

**Note:** Discovery returns all data from the target to the instance before applying console settings. The instance ignores any excluded data when it updates the CMDB.

**Requirements and accessibility**

If you use Internet Explorer, you must use version 8 or later. The configuration console supports keyboard navigation and screen readers.
Console overview

To open the configuration console, navigate to Discovery Definition > Configuration Console.

The console is divided into these sections. Use the toggle buttons to enable various options (right to enable, left to disable).

- **Devices**: network devices such as printers, storage devices such as storage switches, and Unix and Windows computers
- **Applications**: automation applications such as Puppet, databases such as MSSQL, and web servers such as Tomcat
- **Software Filter**: Unix and Windows applications that include or exclude keywords that you enter
- **File-based Discovery**: software files by path, folder, and extensions to scan or ignore
Disabling the discovery of device data at the top level disables the related port probes and pattern libraries. For example, blocking Network Devices disables the SNMP port probe. Blocking devices at the second level of the hierarchy has the following effects:

- Device class categories disable classifiers.
- Device info categories disable probes and pattern libraries.

Device Discovery
Application Discovery

Disabling the discovery of application data affects all host devices on which the application runs. For example, if you configure Discovery to ignore databases, no information is gathered for either Linux or Windows databases. Conversely, if you configure the system to ignore a device type, such as a Windows server, no databases running on that server are explored, even if they are configured to be discovered. The instance cannot identify the applications running on a server until it first discovers that server.

When you exclude an item from the Applications section, the system disables the relevant process classifier. The probes that belong to this classifier, including the Horizontal Pattern probe that launches patterns, never launch.

Software Discovery

You can filter the discovery of software packages based on the keywords they contain for Windows and UNIX computers. A keyword can be any term that is present in a software package name. Keywords for filtering software are stored in the Software Filter Keys (discovery_spkg_keys) table. The system does not provide default filtering keywords for UNIX software.
**Attention:** Keyword software filtering occurs after the probe returns its payload to the instance and not on the target host.

The following default keywords are provided for Windows and are filtered out by default:

- Hotfix
- Language Pack
- Security Update

**File-based Discovery**

You can configure Discovery to either scan or ignore paths, folders, and extensions for File-based Discovery. Discovery then scans a Windows or UNIX file system with these settings to identify file fingerprints. Installed software products detected with these attributes are matched to products in a comprehensive software normalization library.

**Effects on Discovery tables**

This table shows you what is disabled in Discovery tables when you use the Discovery Configuration Console.

<table>
<thead>
<tr>
<th>When you disable</th>
<th>Field disabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top-level device categories, such as:</td>
<td>The <code>Cls</code> field is disabled on the Port probes (discovery_port_probe) table.</td>
</tr>
<tr>
<td>Network Devices</td>
<td>The Port probes (discovery_port_probe) table is referred to from the Protocol Category (discovery_category_protocol) table.</td>
</tr>
<tr>
<td>Windows Servers &amp; Computers</td>
<td></td>
</tr>
<tr>
<td>Unix Servers &amp; Computers</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device classes, such as:</th>
<th>The <code>Active</code> field is disabled on the Classifiers (discovery_classy) table for the related classifier.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 2008 Servers</td>
<td>The Classifiers (discovery_classy) table is referred to from the Device Classification Category (discovery_category_device_class) table.</td>
</tr>
<tr>
<td>Linux Servers</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device info, such as:</th>
<th>The <code>Active</code> field is disabled on both the Classifier Probes (discovery_classifier_probe) and the Discovery Patterns (sa_pattern) tables for the related probe or <code>pattern library</code> that performs the identification and exploration of the CI.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Device Exploration</td>
<td>These tables are referred to from the Device Info Category (discovery_category_device_info) table.</td>
</tr>
<tr>
<td>Windows Storage</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Applications</th>
<th>The <code>Active</code> field is disabled on the Process Classifiers (discovery_classy_proc) table for the related process classifier.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The Process Classifiers (discovery_classy_proc) table is referred to from the Application Category (discovery_category_appl) table.</td>
</tr>
</tbody>
</table>

**Note:** If you use patterns for discovery, and you disable the discovery of device classes or device information, the system disables the pattern libraries that collect the device information. For example, if you have a Windows pattern in which you create a shared...
Exclude CIs from discovery
Use the Discovery Configuration Console to determine which devices and applications to exclude from discovery.

Role required: discovery_admin, admin

1. Navigate to **Discovery > Discovery Definition > Configuration Console**.
2. If you want to exclude specific devices and device information, expand the category for the device types.
3. To see what applications can be disabled, hover your cursor over the application type.
4. Click the toggle button to turn off the CIs you do not want to discover.

The instance creates an update set record for any change you make to the console.

Customize Applications and Devices to Discover
You can toggle on and off Applications and Devices using the Discovery Configuration Console. Some are listed out-of-the-box, but you may also customize which specific Applications and Devices you want to Discover according to your specific requirements.

Role required: admin

1. Customize Applications.
   a) Navigate to **Discovery Definition > Application Categories**.
   b) Click Automation, Databases, or Web & Application Servers.
   c) Unlock the Process classifier list.
   d) To delete from the list, select an application then click the X to remove.
   e) To add to the list, type in the box and click Enter.
   f) Click the lock icon.
   g) Click Update. The next time Discovery runs, it will process those types of specific applications that you configured.

2. Customize Devices.
   a) Navigate to **Discovery Definition > Device Info Categories**.
   b) To create a new record, click New.
   c) Enter the details and click Submit.
   d) To update an existing device, click on a device listed.
   e) Make any changes and click Update or Delete.

Filter software to discover
Use the Discovery configuration console to filter the software you add to the CMDB.

Role required: discovery_admin, admin

The instance creates an update set record for any change you make to the console.

1. Navigate to **Discovery > Discovery Definition > Configuration Console**.
2. In the Software Filter panel, select the tab for the operating system you want, either Unix or Windows.
3. Add or delete keywords for filtering software packages.
To add a keyword, enter it in the field provided and click **New Key**.

To delete a keyword, click the red **x** icon to the right of the keyword.

Discovery looks for software packages containing these terms in the selected operating system and either includes or excludes them from the CMDB. The system does not provide default keywords for filtering UNIX software packages, but provides these default keywords for Windows:

- Hotfix
- Language Pack
- Security Update

4. Select whether to **Include** or **Exclude** software packages with the keywords in their names.
   - **Include**: Only add software packages that match the keywords.
   - **Exclude**: Add all software packages except those matching the keywords.

5. Disable software filtering for either **UNIX** or **Windows**.

   When the filter is disabled for an operating system, all discovered software packages for that operating system are added to the CMDB, with no filtering applied.

   The instance creates an update set record for any change you make to the console.

**Configure File-based Discovery**

Use the Discovery configuration console to configure several settings to find and manage all of your installed software.

Role required: discovery_admin, admin

1. Navigate to **Discovery > Discovery Definition > Configuration Console**.
2. Enable **File-based Discovery**.
   - This is disabled by default.
3. Select customizable options specifically for Windows and UNIX machines as well as options common to both types.
   - Some options have default values and others, such as folders to ignore or extensions, are blank until you add in values.
4. For Windows and/or UNIX, click the arrow to expand the target path options.
5. Configure basic selections.

**Folders to Scan**

Indicates which folders to scan for Windows and/or UNIX. Default values are listed. You can also add additional paths to scan.
Folders to Ignore

Indicates which folders and child folders/files to ignore from within the list of Window or UNIX folders that you want to scan. For example, if you have specified “C:\Program Files” as a folder to scan, you could additionally specify “C:\Program Files\Temp” as an ignore folder. When File-based Discovery scans a Windows server, it will scan all files and sub folders under C:\Program Files, excluding C:\Program Files\Temp.

Extensions Blacklist

Indicate extensions (.txt, .war, .exe, etc.) that you want to ignore for Windows.

Extensions Wildcards

Indicate extensions (.cert, .crt, etc.) that you always want to scan for Windows. This allows you to scan for extensions such as *.cert where the file name itself might not be known ahead of time. You should be able to find certificates (.cert or .crt) and java keystore files (.jks) in this way.

**Advanced**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sleep Time</td>
<td>10000</td>
</tr>
<tr>
<td>File Throttle</td>
<td>500</td>
</tr>
<tr>
<td>Skip Hidden Folders</td>
<td></td>
</tr>
<tr>
<td>Scan Running Process Paths</td>
<td></td>
</tr>
<tr>
<td>Scan SWID Tags</td>
<td></td>
</tr>
</tbody>
</table>

**Sleep Time**

Sets the length of time in milliseconds to sleep during each throttling interval while scanning for Windows files. The default value is 10000. Do not change this value unless you are an advanced user. If you are an advanced user and need to make a change, make sure to set to a non-zero value.

**File Throttle**

Sets the number of files to scan on Windows before sleeping. The default value is 500. Do not change this value unless you are an advanced user. If you are an advanced user and need to make a change, make sure to set to a non-zero value.

**Skip Hidden Folders**

Skips hidden folders on target computers when scanning for file signatures. For Windows devices, the visibility of a folder is based on a hidden attribute that can be set for each folder.

**Scan Running Process Paths**

Enables the use of running process paths in the list of targeted paths to scan. This automatically adds paths that we know contain running processes and therefore most likely contain files of interest for use in determining software installations.
Scan SWID Tags

Scans for and analyzes the software identification (SWID) tags. This option is only visible if SAMP is enabled. When Scan SWID Tags is enabled, this returns the software product, publisher, and version in the payload to be directly inserted into software installs table, bypassing the software match API. This is disabled by default.

7. Configure **Advanced UNIX** settings.

![Advanced settings](image1)

Skip Hidden Folders

Skips hidden folders when scanning for file signatures. For Unix-based devices these are folders prefixed by a “.” in their name.

Scan Running Process Paths

Enables the use of running process paths in the list of targeted paths to scan. This automatically adds paths that we know contain running processes and therefore most likely contain files of interest for use in determining software installations.

8. Configure **Common** settings for either Windows or UNIX.

![Common settings](image2)

Maximum Files

Sets the maximum number of files that you want to discover. This is the number of files scanned per CI/IP.

**Frequency**

Sets the frequency that File-based Discovery runs on the CI.

File-based Discovery is now enabled, configured, and ready for you to run. See [Run File-based Discovery](#) for more information.

**Schedule a horizontal discovery**

A discovery schedule determines what horizontal discovery searches for, when it runs, and which MID Servers are used. Create a Discovery schedule for your local environment or a schedule for discovering the resources in your cloud service account.
Roles required: admin, discovery_admin

You can use a Discovery schedule to launch horizontal discovery, which uses probes, sensors, and pattern operations to scan your network for CIs. Use this procedure to create a schedule manually from the Discovery Schedule form.

Service Mapping also provides a Discovery schedule for top-down discovery. See Schedule a top-down discovery by Service Mapping for more information.

Use the Discovery Schedule module in the Discovery application to:

- Configure a schedule to discover resources in your cloud service account.
- Configure device identification by IP address or other identifiers.
- Determine if credentials are used in device probes.
- Name the MID Server to use for a particular type of discovery.
- Create or disable a schedule that controls when the discovery runs in your network.
- Configure the use of multiple Shazzam probes for load balancing.
- Configure the use of multiple MID Servers for load balancing.
- Run a Discovery schedule manually.
- Run Discovery on a single IP address.

**Note:** To view the run-results of your schedules for both IP-based and Cloud Discovery, use the summaries on the Discovery Home page. The Home page publishes the details of any errors that might have occurred and displays possible actions to take to remediate problems.

1. Navigate to Discovery > Discovery Schedules to create a new record.
2. Select the type of schedule to open:
   - **New:** Creates a new horizontal schedule for discovering components in your network.
   - **Quick Discovery:** Runs an horizontal discovery on a single IP address without requiring a schedule.
   - **Create a Cloud Discovery schedule:** Creates a schedule, using the Discovery Manager wizard, for discovering resources in a cloud service account.
3. Complete the Discovery Schedule form, using the fields in the table.
4. Right-click in the header of the record and select Save from the context menu.
5. To create a range of IP addresses to discover, click Quick Ranges under Related Links.
## Discovery Schedule

**Name:** My Discovery Schedule

**Discover:** Configuration Items

**NIS Server selection method:** Auto-Select NIS Server

**Max run time:**
- Days: 0
- Hours: 10
- Minutes: 00

**Run:** Weekly

**Day (run, day/week):** Monday

**Time:**
- Hours: 02
- Minutes: 00

### Advanced

- **Include alive:** Yes
- **Hashcheck size:** 5000
- **Log state changes:** Yes
- **Hashcheck cluster support:** Yes

### Related Links

**Discovery IP Ranges**

<table>
<thead>
<tr>
<th>Type</th>
<th>IP Network</th>
<th>Summary</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.1.0.0/16</td>
<td></td>
<td>true</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Discovery Schedule Form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique, descriptive name for your schedule.</td>
</tr>
<tr>
<td>Discover</td>
<td>Select one of the following scan types:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Configuration items</strong>: Uses Discovery identifiers to match devices with CIs in the CMDB and update the CMDB appropriately. Perform a simple discovery by selecting a specific MID Server to scan for all protocols (SSH, WMI, and SNMP). Or, perform advanced discoveries with discovery behaviors. When you select a behavior, the MID Server field is not available.</td>
</tr>
<tr>
<td></td>
<td>• <strong>IP addresses</strong>: Scans devices without the use of credentials. These scans discover all the active IP addresses in the specified range and create device history records, but do not update the CMDB. IP address scans also show multiple IP addresses that are running on a single device. Identify devices by class and by type, such as Windows computers and Cisco network gear. The Max range size Shazzam probe property determines the maximum number of IP addresses Shazzam scans. See <a href="#">Configure Shazzam probe</a> for details.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Networks</strong>: Discovers IP networks (routers and switches). Results from this search are used to populate the IP Network (cmdb_ci_ip_network) table in Discovery &gt; IP Networks with a list of IP addresses and network masks. Network scans update routers and layer 3 switches in the CMDB.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Service</strong>: Discovers services for the Service Mapping application. See <a href="#">Schedule a top-down discovery by Service Mapping</a> for instructions.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Serverless</strong>: Finds CIs without needing to run discovery on a host, or CIs on a proxy host that is already in the CMDB. See <a href="#">Serverless Discovery</a> for more information.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Cloud application</strong>: Discovers only the cloud resources for the patterns that you specify. See <a href="#">Cloud Discovery</a> for instructions.</td>
</tr>
</tbody>
</table>
|           | • **Cloud resources**: Discovers resources on AWS and Azure clouds. This option only appears when you run Discovery on a [cloud service account](#). You cannot select it from a new Discovery schedule.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MID Server selection method</td>
<td>Select the method that Discovery uses to select a MID Server:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Auto-Select MID Server</strong>: Allow Discovery to select the MID Server automatically based on the Discovery IP Ranges you configure. To find a matching MID Server, you must configure MID Servers to use:</td>
</tr>
<tr>
<td></td>
<td>• The Discovery application, or ALL applications. This setting authorizes the MID Server access from Discovery.</td>
</tr>
<tr>
<td></td>
<td>• The IP Range that includes the ranges you configure on the Discovery Schedule.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Specific MID Cluster</strong>: Use a preconfigured cluster of MID Servers. Select the cluster. You are not required to specify one member of the cluster. The MID Server cannot be part of multiple clusters, such as one that supports load balancing and one that supports failover. You can add any cluster regardless of the application that the MID Servers are assigned to. When you select the cluster, the <strong>Discovery</strong> application is automatically added when it does not exist for the MID Servers in the cluster.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Specific MID Server</strong>: Use only one MID Server. If that MID Server is part of a cluster, only that MID Server is used. The cluster is not used. You can add any MID Server regardless of the application it is assigned to. The <strong>Discovery</strong> application is automatically added when it is not already assigned for the MID Server you select. You can assign a specific MID Server for all types of Discover scans except <strong>Service</strong>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Use Behavior</strong>: Use a behavior when a single schedule requires the use of multiple MID Servers to perform any of the following activities:</td>
</tr>
<tr>
<td></td>
<td>• Scans requiring multiple Windows credentials.</td>
</tr>
<tr>
<td></td>
<td>• A schedule that must execute two or more particular protocols (SNMP, SSH, or WMI) using more than one MID Server.</td>
</tr>
<tr>
<td></td>
<td>• Load balancing for large discoveries where a single MID Server would be inadequate.</td>
</tr>
<tr>
<td></td>
<td>• Scanning multiple domains.</td>
</tr>
</tbody>
</table>

**Note**: The Discovery schedule enforces domain separation. The MID Servers that are available for selection are limited to the same domain of the user who is configured.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| MID Server          | Select the MID Server to use for this schedule. This field is available if MID Server selection method is set to Specific MID Server, or if you discover IP addresses, networks, or web services.  
To verify that the MID Server you selected is up and validated, look at the [MID Server dashboard](#). |
| MID Server Cluster  | Select the MID Server cluster to use for this schedule. This field is available if MID Server selection method is set to Specific MID Cluster.                                                                   |
| Behavior            | Select a behavior configured for the MID Servers in your network.  
This field is available only if MID Server selection method is set to Use Behavior.                                                                                                           |
| Active              | Select the check box to enable this schedule. If you clear the check box, the schedule is disabled, but you can still run a discovery manually from this form, using the configured values.                               |
| Location            | Choose a location to assign to the CIs that the schedule discovers. If this field is blank, then no location is assigned.                                                                                       |
| Max run time        | Set a time limit for running this schedule. When the configured time elapses, the remaining tasks for the discovery are canceled, even if the scan is not complete.  
Use this field to limit system load to a desirable time window. If no value is entered in this field, this schedule runs until complete.                |
| Run and related fields | Determines the run schedule of the discovery. Configure the frequency in the Run field and the other fields that appear to specify an exact time.  
**Note:** The run time always uses the system time zone. If you add the optional Run as tz field, it has no effect on the actual runtime.            |
| Include alive       | Select this check box to include alive devices, which are devices that have at least one port that responds to the scan, but no open ports. Discovery knows that there is a device there, but has no information about it.  
If this check box is cleared, Discovery returns all active devices, which are devices that have at least one open port. |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log state changes</td>
<td>Select this check box to create a log entry every time the state changes during a discovery, such as a device going from Active to Classifying. View the discovery states from the Discovery Devices related list on the Discovery Status form. The Completed activity and Current activity fields display the states.</td>
</tr>
<tr>
<td>Shazzam batch size</td>
<td>Enter the number of IP addresses that each Shazzam probe can scan. Dividing the IP addresses into batches improves performance by allowing classification for each batch to begin after the batch completes, rather than after all IP addresses have been scanned. The probes run sequentially. For example, the value is set to 1000 and a discovery scans 10,000 IP addresses using a single MID Server. It creates 10 Shazzam probes with each probe scanning 1000 IP addresses. By default, the batch size is 5000. A UI policy enforces a minimum batch size of 256 because batch sizes below 256 IP addresses do not benefit from clustering. The policy converts any value below 256 to a value of zero. The value for this field cannot exceed the value defined in the maximum range size property for the Shazzam probe.</td>
</tr>
<tr>
<td>Shazzam cluster support</td>
<td>Select the check box to distribute Shazzam processing among multiple MID Servers in a cluster and improve performance. This setting works with the Shazzam batch size. For example, a schedule is created to scan 100,000 IP addresses, with 10 MID Servers assigned to do the work. Each MID Server is assigned to scan 10,000 IP addresses. If the Shazzam batch size is set to 5,000 IP addresses per probe, the schedule runs two Shazzam probes per MID Server (10,000 IP addresses/5,000 per batch). These probes are run in sequence and not concurrently.</td>
</tr>
<tr>
<td>Use SNMP Version</td>
<td>Use this field to designate the SNMP version to use for this discovery. Valid options are v1, v2c, v3, or All.</td>
</tr>
<tr>
<td>Quick ranges</td>
<td>Define IP addresses and address ranges to scan by entering IP addresses in multiple formats (network, range, or list) in a single, comma-delimited string. For more information, see Create a Quick IP range for a Discovery schedule.</td>
</tr>
<tr>
<td>Discover now</td>
<td>Use this link to immediately start this Discovery.</td>
</tr>
</tbody>
</table>
### Discovery schedule run options

<table>
<thead>
<tr>
<th>Run option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>On Demand</td>
<td>Does not run on a schedule. Click <strong>Discover now</strong> to run Discovery. This is the default option.</td>
</tr>
<tr>
<td>Daily</td>
<td>Runs every day. Use the <strong>Time</strong> field to specify the time of day.</td>
</tr>
<tr>
<td>Weekly</td>
<td>Runs on one designated day of each week. Use the <strong>Time</strong> field to specify the time of day.</td>
</tr>
<tr>
<td>Monthly</td>
<td>Runs on one designated day of each month. Use the <strong>Day</strong> field to select the designated day and the <strong>Time</strong> field to specify the time of day. If the designated day does not occur in the month, the schedule does not run. For example, if you designate day 30, the schedule does not run in February.</td>
</tr>
<tr>
<td>Periodically</td>
<td>Runs every designated period of time. Use the <strong>Repeat Interval</strong> field to define the period in days, hours, minutes, and seconds. The first Discovery runs at the point in time defined in the <strong>Starting</strong> field. Discovery runs after each <strong>Repeat Interval</strong> period passes.</td>
</tr>
<tr>
<td>Once</td>
<td>Run one time as designated by the date and time defined in the <strong>Starting</strong> field.</td>
</tr>
<tr>
<td>Weekdays</td>
<td>Runs every Monday, Tuesday, Wednesday, Thursday, and Friday. Use the <strong>Time</strong> field to select the time of day.</td>
</tr>
<tr>
<td>Weekends</td>
<td>Runs every Saturday and Sunday. Use the <strong>Time</strong> field to select the time of day.</td>
</tr>
<tr>
<td>Month Last Day</td>
<td>Run the last day of every month. Use the <strong>Time</strong> field to select the time of day.</td>
</tr>
<tr>
<td>Calendar Quarter End</td>
<td>Runs on March 31, June 30, September 30, and December 31. Use the <strong>Time</strong> field to select the time of day. To change the dates, modify the DiscoveryScheduleRunType script included.</td>
</tr>
<tr>
<td>After Discovery</td>
<td>Allows you to sequentially stagger the schedule. Use this option to run this schedule after the <strong>Run after</strong> Discovery finishes. Select the <strong>Even if canceled</strong> check box to designate the Discovery is canceled before it finishes.</td>
</tr>
<tr>
<td></td>
<td>- This option is not valid when the Discovery is started via DiscoverNow, or if you cannot designate an inactive Discovery schedule.</td>
</tr>
<tr>
<td></td>
<td>- You cannot create a loop by designating the run after Discovery to be the same Discovery.</td>
</tr>
<tr>
<td></td>
<td>- This Discovery does not run if the <strong>Run after</strong> Discovery does not finish, with the exception that the <strong>Even if canceled</strong> check box is selected and the Discovery is canceled.</td>
</tr>
</tbody>
</table>

---

**Run a Quick Discovery**

Quick Discovery, or DiscoverNow, allows an administrator to run a CI Configuration discovery on a single IP address without requiring a schedule.

The platform automatically selects the correct MID Server to use for the discovery if one is associated with the IP address selected. If no MID Server is configured for the network in which that address appears, you can select a MID Server. Use this feature to discover new devices in the network as soon as they are connected to the network, rather than waiting for a regularly scheduled discovery.
To configure the system to automatically determine which MID Server to use, set up the IP range capabilities for each MID Server in your system.

You can run DiscoverNow from a Discovery schedule form or from a script.

1. Open Quick Discovery from one of these locations:
   - Navigate to Discovery > Discovery Schedules and click Quick Discovery in the header bar.
   - Navigate to Discovery > Home and click Discovery Quick Start under the Schedules tile.

   A dialog box appears asking for an IP address and the name of the MID Server to use. Only Up and Validated MID Servers are available.

2. Enter the target IP address for a discovery in the Target IP field.

   \[\textbf{Note:}\] DiscoverNow does not currently support IP network discovery. Make sure that you enter a single IP address only and not an entire network, such as 10.105.37.0/24.

   When a MID Server is assigned to the subnet containing the target IP address and currently in an operational status of Up, the name appears automatically in the MID Server field. If multiple MID servers are found, the system selects one for you. The value in the MID Server field can be overwritten if you want to select a different MID Server.

   \[\textbf{Attention:}\] If the selected MID Server is part of a load balanced cluster and becomes unavailable for any reason, the instance does not assign another MID Server from that cluster to the quick Discovery. You must select another MID Server from the list of appropriate MID Servers.

3. If no MID Server is defined for that network, select one from the list of available MID Servers.

   \[\textbf{Quick Discovery Dialog}\]

4. Click OK to run discovery.

   The status record for that discovery appears. The Schedule column is empty because no schedule is associated with this discovery.
Quick Discovery Status List

Run DiscoverNow from a script
You can run DiscoverNow from a script, such as a background job, a business rule, or web services.
Role required: admin

1. Create the following script:

   ```javascript
   var d = new Discovery();
   var statusID = d.discoveryFromIP(TARGET_IP, TARGET_MIDSERVER);
   ```

   The `discoveryFromIP` method takes two arguments: IP and MID Server. The IP argument is mandatory, but the MID Server argument is optional.

2. To choose the MID Server, supply either the `sys_id` or name of the MID Server as the argument. If you do not name a MID Server, the system attempts to find a valid one automatically. A valid MID Server has a status of Up and can discover the given IP address. If the system finds a valid MID Server and runs a Discovery, the `discoveryFromIP` method returns the `sys_id` of the Discovery status record. If no MID Server can discover this IP address, the method returns the value `undefined`.

   If you manually specify the `TARGET_MIDSERVER`, the system validates the given value and ensures that the MID Server table contains the specified MID Server record. If the validation passes, the `discoveryFromIP` method returns the `sys_id` of the discovery status record. If the validation fails, the method return the value `undefined`.

Validate discovery results
Validate the results of your discovery by accessing the ECC queue, analyzing the XML payload, and checking the Discovery log.
Role required: discovery_admin

Initial discoveries often reveal unexpected results, such as previously unknown devices and processes or failed authentication. Results should also accurately identify known devices and update the CMDB appropriately. Become familiar with the network that is being discovered and the types of data returned for the different types of discoveries. Use the Discovery Log and the ECC Queue to monitor the Discovery process as data is returned from probes or pattern operations.

1. To view the actual payload of a probe, click the XML icon in a record in the ECC Queue.
This XML file does not appear to have any style information associated with it. The document tree is shown below.

```xml
<parameters>
  <parameter name="port" value="5988"/>
  <parameter name="source" value="10.1.134.226"/>
  <parameter name="device_instance" value=" Hitachi StorageVolume [Name: AN20100.03043211, CreationClassName: Hitachi_StorageSystem, Device ID: 8D04321120022, CreationClassName: Hitachi_StorageVolume, SystemName: AN20100.03043211] " />
  <parameter name="probe" value="CIM - SNIA - Storage Controllers" />
  <parameter name="queries" value="(Device[Name: AN20100.03043211, CreationClassName: Hitachi_StorageSystem, Device ID: 8D04321120022, CreationClassName: Hitachi_StorageVolume, SystemName: AN20100.03043211])" />
  <parameter name="credential_id" value="79fe74a9373110004d0fe5d12" />
</parameters>
```
2. To view the actual payload of a probe, click the XML icon in a record in the ECC Queue.

3. Use the Discovery Log form for a quick look at how the probes are doing. To display the Discovery Log, navigate to Discovery > Discovery Log.

### Discovery Log

The Discovery Log provides this information:

<table>
<thead>
<tr>
<th>Column</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created</td>
<td>Displays the timestamp for the probe launched. Click this link to view the record for the probe launched in this list.</td>
</tr>
<tr>
<td>Level</td>
<td>Displays the type of data returned by this probe. The possible levels are:</td>
</tr>
<tr>
<td></td>
<td>- Debug</td>
</tr>
<tr>
<td></td>
<td>- Error</td>
</tr>
<tr>
<td></td>
<td>- Information</td>
</tr>
<tr>
<td></td>
<td>- Warning</td>
</tr>
<tr>
<td>Message</td>
<td>Message describing the action taken on the information returned by the probe.</td>
</tr>
<tr>
<td>ECC queue input</td>
<td>Displays the ECC queue name associated with the log message.</td>
</tr>
<tr>
<td>CI</td>
<td>The CI discovered. Click this link to display the record from the CMDB for this CI.</td>
</tr>
<tr>
<td>Source</td>
<td>Displays the probe name that generated the log message.</td>
</tr>
</tbody>
</table>
### Column Information

| Device | Displays the IP address explored by the probe. Click this link to examine all the log entries for the action taken on this IP address by this Discovery. |

**Note:** If you cancel an active discovery, note the following information:
- Existing sensor jobs that have started processing are immediately terminated.
- The existing sensor jobs that are in a Ready state, but have not started processing, are deleted from the system.

4. View the [Discovery Home page](#) for details about all schedules, cloud resources (virtual machines), discovered devices, and related errors that might have occurred. **Error details** include possible remediation steps.

### MID Server selection sequence for Discovery schedules

The Discovery application follows this sequence to find a MID Server.

#### MID Server auto-selection

Discovery follows this sequence when you select **Auto-Select MID Server** for the **MID Server selection method** on the Discovery Schedule form.

1. Discovery looks for a MID Server that also has an appropriate IP range configured.
2. If no MID Servers meet these criteria, it looks for a MID Server that has the **ALL** application that also has an appropriate IP range configured.
3. If more than one MID Servers meet the criteria, Discovery chooses the first MID Server with the status of **Up**. If more than one MID Servers are up, it randomly picks one.
4. If none are up, it uses the default MID Server specified for the Discovery application, assuming it is up.
5. If no default MID Server is specified, it uses the default MID Server specified for the **ALL** application, assuming it is up.
6. If no default MID Server is specified, Discovery cycles through the previous steps and looks for MID Servers with the status of **Paused** or **Upgrading**.

**Note:** When a MID Server is paused or upgrading, it does not actually process commands until it returns to the status of **Up**.

### MID Server clusters

These steps are followed when you select **Specific MID Cluster** for the **MID Server selection method** on the **Discovery** form, and the cluster is a load balancing cluster:

1. **Discovery** uses the first MID Server in the cluster that it finds with the status of **Up**.
2. If more than one MID Servers are up, it randomly picks one. If it cannot find any MID Servers, it looks for MID Servers in the cluster with the status of **Paused** or **Upgrading**.

These steps are followed when the cluster is a failover cluster:

1. **Discovery** uses the MID Server with the lowest **Order** value that also has the status of **Up**.
2. If no MID Servers are found, it looks for MID Servers in the cluster with the status of **Paused** or **Upgrading**, choosing the one with the lowest **Order** value.

**Note:** **Discovery** ignores the default MID Server for it and **ALL** applications when selecting a MID Server from the cluster.

**Port scan (Shazzam) phase**

During the port scan phase, **Discovery** collects all the target IP addresses. It splits them equally between MID Servers matching the criteria (MID Servers are qualified to do the port scan). The Shazzam batch size, which you configured on the **Discovery** schedule, determines the number of IP addresses that each Shazzam probe can scan. This phase helps determine how much work each MID Server does during the port scan phase.

For example, you have 16,000 IP addresses to scan among three qualified MID Servers, and you use the default Shazzam batch size of 5000. Two of the MID Servers handle 5000 IP address scans (one Shazzam probe each). The other MID Server handles 6000 IP address scans by launching two Shazzam probes.

**File-based Discovery**

File-based Discovery helps you identify what software is running on your Windows and UNIX servers and devices, even if there is no registration information available. You can then manage and maintain records of your software licenses, check for unlicensed files, detect forbidden or damaged files, and help evaluate any threats from unwanted files.

**Required plugins**

The File-based Discovery (com.snc.discovery.file_based_discovery) plugin is required for file signature filtering. Your Discovery subscription includes this plugin, but you must request activation. Once the File-based Discovery plugin is active, the Software Asset Management - File Signature Normalization (com.snc.file_signature_normalization) plugin will also be activated. For more information on the File Signature Normalization plugin, see [File Signature Normalization](#).

**How File-based Discovery works**

File-based Discovery enhances the pre-existing discovery of installed software. It scans target servers for a known list of file signatures and processes those files with an established set of rules. The resulting data enhances the identification of installed software and identifies unregistered software products.

File-based Discovery is triggered in the exploration phase of normal Discovery. File-based Discovery probes execute a scan searching for specific file extensions or file names in paths that you configure. The resulting file information is returned in the probe payload. The sensor attempts to match the discovered files with installed software, using the file name, size, and version returned by the probe. File-based Discovery uses file signatures to detect software that might not have
been registered. This information is then stored in the File Information (cmdb_file_information) table with a reference to the CI of the server. You can view the files found from each CI in a related list on this table. For more information, see Related Lists of CI components.

When Software Asset Management Professional (SAMP) is active, if any file matches a software product, Discovery populates the Product and Publisher information for that file. Use this information to understand what software is running on your server and to help evaluate any threats from unwanted files. Discovery uses lists of known file signatures for Windows and UNIX to constrain the scope of the search. The filtering process for Windows and UNIX hosts is executed differently because their signature lists differ greatly in size. The much smaller UNIX signature list is included with the Unix - File Discovery probe and processed directly on the target. The Windows signature list is much larger and cannot be processed on the target. The Windows - File Discovery probe scans the target for specific file extensions and paths and returns these results to the MID Server. The MID Server performs file signature filtering using the entire Windows list. The MID Server then sends all file information back to the instance for normalization and matching.

If SAMP is active on the instance, File-based Discovery creates or updates identified software products in the Software Installation (cmdb_sam_sw_install) table and updates the licenses of matched software packages. Without SAMP, no software records are created and only the file information goes into the File Information (cmdb_file_information) table.

File-based Discovery filtering flow

File-based Discovery inserts any file not matched by the normalization API into the Unidentified File Set (cmdb_unidentified_file_set) table. You can update the records in this table and provide additional details for previously unidentified files. If you provide values for the Product and Publisher fields for a file, settings in SAMP can enable File-based Discovery to use that file for installed software matching in future discoveries.
**Note:** You can disable File-based Discovery at any time by changing the setting in the Discovery Configuration Console. If you disable File-based Discovery before scan results are returned, the file data is ignored.

File-based Discovery table schema

**Note:** File-based Discovery supports Windows and UNIX devices. The UNIX probe is POSIX-compliant and should run on any Linux/Solaris server. We support Windows versions 2008, 2008R2, 2012R2, 2016, 2019, and above with PowerShell 3.0 and above. We also support AIX versions 5.3, 6.1, and 7.1 and HP/UX 8.11.
1. **File-based Discovery reference information**

*Run File-based Discovery*
Run File-based Discovery to find all of your installed software whether it is registered or not. You can enable and configure File-based Discovery at any time using the Discovery Configuration Console.

Role required: admin

1. Set up the PowerShell script.
   
   By default, the `filebaseddiscovery.ps1` script has a ServiceNow signature. Its certificate chain resolves up to the VeriSign Universal Root Certification Authority which is trusted by Windows by default. Since PowerShell scripts are signed by ServiceNow publisher, add ServiceNow publisher to your trusted publisher repository.

   ![](image)

   If your Execution Policy requires you to use your own certificate or if you need to make any changes to the script, re-sign the script.
1. Navigate to the probe “Windows - File discovery.”
2. Open the filebaseddiscovery.ps1 probe parameter.
3. Copy the contents of the value field into a file.
4. Make the necessary changes to the file and then remove the (old) signature block at the end of the file.
   After re-signing, the file should have a new signature block at the end of it.
6. Copy the entire file contents and replace the old value in the filebaseddiscovery.ps1 probe parameter from step b.
   Windows File-based Discovery should now run using the updated signed script.

If you are using anti-virus software, make sure to whitelist all of the ServiceNow Windows PowerShell scripts so the anti-virus software does not see it as a threat.
2. Activate the File-based Discovery plugin.
3. Enable and configure File-based Discovery.
   Using the Discovery Configuration Console, you can enable File-based Discovery as well as configure several components to find and manage all of your installed software.
   To avoid impacting performance for existing customers, File-based Discovery is disabled by default.
4. Configure File-based Discovery scans.
   After Discovery runs and returns file information for a CI, it will not execute File-based Discovery again on that target until the interval has expired. Since there is a performance cost when File-based Discovery performs scans, it is important to determine how frequently to scan. However, choosing a more frequent interval than Monthly is not recommended due to performance considerations.
   a) Using the Discovery Configuration Console, expand Common and then enter the maximum number of files that you want to discover. NOTE: Increasing this value can impact performance.
   b) Select the Frequency that File-based Discovery runs on the CI. The default is set to Monthly.

   Note: File-based Discovery does not trigger until initial Content Data Service (CDS) synchronization occurs. CDS synchronization could take approximately 24 hours for the initial set of data to be synchronized from CDS. For more information see, File Signature Normalization.

Your File-based Discovery should run based on the configurations set. You can then monitor the results using the Discovery Dashboard.

1. File-based Discovery reference information

File-based Discovery issue resolution
If you have any issues while setting up or running File-based Discovery, follow the actions listed here to help resolve them.
<table>
<thead>
<tr>
<th>Issue</th>
<th>Cause/Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error message: File Discovery data not yet synced...</td>
<td>• Allow CDS time to sync the samp_file_* tables.</td>
</tr>
<tr>
<td></td>
<td>• Check cds_client_schedule table and see if timestamp has been updated for</td>
</tr>
<tr>
<td></td>
<td>&quot;samp_file_*&quot; tables.</td>
</tr>
<tr>
<td></td>
<td>File-based Discovery is not triggered during Discovery.</td>
</tr>
<tr>
<td>File-based Discovery is not triggered during Discovery.</td>
<td>• Feature has not been enabled. Check the Discovery Definition Configuration</td>
</tr>
<tr>
<td></td>
<td>Console and ensure that File-based Discovery is enabled.</td>
</tr>
<tr>
<td></td>
<td>• Interval between scans is too long. Check the interval setting on the</td>
</tr>
<tr>
<td></td>
<td>Discovery Definition Configuration Console.</td>
</tr>
<tr>
<td></td>
<td>Check the Internal properties managed by the Discovery</td>
</tr>
<tr>
<td></td>
<td>(discovery_private_properties) table for the following items. To view</td>
</tr>
<tr>
<td></td>
<td>this table, enter discovery_private_properties.list in the navigation</td>
</tr>
<tr>
<td></td>
<td>filter and press Enter.</td>
</tr>
<tr>
<td></td>
<td>• File_discovery.file_extensions_windows should contain a list of Windows</td>
</tr>
<tr>
<td></td>
<td>file extensions to be scanned.</td>
</tr>
<tr>
<td></td>
<td>• File_discovery.unix_filename_timestamp should be something other than</td>
</tr>
<tr>
<td></td>
<td>1970 timestamp.</td>
</tr>
<tr>
<td></td>
<td>• File_discovery.windows_filename.timestamp should be something other than</td>
</tr>
<tr>
<td></td>
<td>1970 timestamp.</td>
</tr>
<tr>
<td>Errors in MID Server log from FileDiscoveryFilter or MID Server</td>
<td>• Look to see if the File-based discovery</td>
</tr>
<tr>
<td>issue created.</td>
<td>whitelist exists. The whitelist is in the MID Server</td>
</tr>
<tr>
<td></td>
<td>installation folder in /agent/work. If it does not exist, the sync did</td>
</tr>
<tr>
<td></td>
<td>not succeed.</td>
</tr>
<tr>
<td></td>
<td>• Look at the MID Server record on the instance to see if a MID Server</td>
</tr>
<tr>
<td></td>
<td>issue was created.</td>
</tr>
<tr>
<td></td>
<td>• MID Server uses the GetMIDInfo scripted soap service to synchronize the</td>
</tr>
<tr>
<td></td>
<td>whitelist. See if errors are generated from this script.</td>
</tr>
<tr>
<td></td>
<td>• Click the File discovery filter refresh Related Link in the MID Server</td>
</tr>
<tr>
<td></td>
<td>record to synchronize the file names to this MID Server.</td>
</tr>
<tr>
<td>PowerShell errors</td>
<td>• Whitelist the script in your anti-virus software.</td>
</tr>
<tr>
<td></td>
<td>• Add ServiceNow publisher to your trusted publisher repository.</td>
</tr>
</tbody>
</table>

1. **File-based Discovery reference information**

**File-based Discovery references**

Review this reference material for further details on File-based Discovery.
How File-based Discovery minimizes performance impact

Scanning large networks, particularly for Windows computers, can extend scan time considerably. Moreover, file scanning can create heavy CPU demands on target machines and produce large payloads that greatly increase processing time. To avoid performance issues, File-based Discovery allows you to optimize processing with many of the settings configured using the Discovery Configuration Console.

- **Multi-phase scan**: Probes are triggered in multiple phases to reduce long scan times. The first probe creates the scanning process on the target, which runs in the background. Subsequent probes check the progress of this scan each time Discovery runs and return the results when the scan is complete. For example, if a file scan is initiated via Quick Discovery or from a Discovery schedule, no information is returned if a background scan is still running. If the scan is finished the next time the server is discovered, the probe returns the data.

  **Attention**: The multi-phase scan that File-based Discovery uses does not return results immediately. Do not expect any results from a file-based scan until Discovery runs again on those targets and fetches the list of files discovered by the previous scan.

- **Scan folders**: You can limit the scope of the scan to particular folders as well as scan hidden folders.
- **Scan extensions**: You can retrieve all files for a given list of extensions as well as ignore specific file extensions when scanning.
- **Scan UNIX file names**: Because the number of unique file names used to identify installed software on UNIX is relatively small, the complete list of file names is included in the UNIX probe. This enables the probe to filter the files it discovers against the signature list on the target. No additional filtering is required until the results are returned to the instance.
- **Scan Windows file names**: For Windows File-based Discovery, the probe does not initiate a scan on target for full signature list. Instead, Discovery generates a list of file extensions that the probe attempts to match on the target. This reduced list is then filtered on the MID Server against the entire Windows signature list.
- **Reduced payloads**: By filtering the scan results with the file signature list before the results get to the instance, Discovery reduces the size of the payload from target servers. Only those files from the targets that match the files specified in this list are returned to the instance.
- **Scan throttling**: To reduce the impact on Windows CPU and I/O performance on remote targets, Windows File-based Discovery is configured to sleep occasionally during scans. You can fine-tune this behavior with Discovery properties. Throttling Windows scans can make those scans take longer.

File-based Discovery tables

Descriptions of the tables provided by the File-based Discovery (com.snc.discovery.file_based_discovery) plugin for storing data:

### File Information (cmdb_file_information)

This table stores all the files, matched or unmatched, that were discovered on a particular CI. The files are available in a related list on CI computer records. The following columns are provided in the default view:

- **Path**: Path in which the file was found on the target.
- **Product**: Software product name. This field is populated when SAMP is activated and a match exists.
• **Publisher**: Name of the publisher of the file, such as Oracle or Microsoft. This is a reference to the Company (core_company) table. This field is populated when SAMP is activated and a match exists.

• **Size**: Size of the file.

• **Software installation**: Name of the installed software associated with this file, from the Software Installation (cmdb_sam_sw_install) table. This field is populated when SAMP is activated and a match exists.

• **Version**: Version of the file.

**Unidentified File Set (cmdb_unidentified_file_set)**

This table stores all the files returned by Discovery that cannot be matched to a software product, with or without SAMP. Discovery populates the name, size, and version of each unmatched file in the record. These optional fields can be used to further identify the file and make it available for future file matching:

- Publisher
- Product
- Version
- Edition
- Platform
- Language

**Note**: A feature in Software Asset Management Professional (SAMP) allows signature data in this table to be shared with Content Data Services (CDS). When you enable data sharing, the system automatically uploads records from the Unidentified File Set (cmdb_unidentified_file_set) table to CDS when you provide the missing information. Uploading the identified files makes them available to improve future file matching. For details, see *Add a custom software product*.

**File-based Discovery troubleshooting**

If you are experiencing any issues when setting up or running File-based Discovery, see *File-based Discovery issue resolution*.

**Network discovery**

Network Discovery discovers the internal IP networks and subnets within your organization.

**Note**: If you already know the IP address ranges in your network, it is not necessary to run Network Discovery. It is intended for organizations that do not have complete knowledge of the IP addresses available for Discovery in their networks.

**How network discovery works**

Discovery uses the information it gathers to update routers and Layer 3 switches in the CMDB. A single MID Server that begins its scan on a configurable list of starting (or seed) routers performs network Discovery. Typically, the starting routers are the default routers used by all the MID Server host machines in the network, but can be any designated routers. The MID Server uses the router tables on the starting routers to discover other routers in the network. The MID Server then spreads
out through the network, using router tables it finds to discover other routers. It continues until all the routers and switches have been explored.

After running Network Discovery, convert the IP networks it finds into IP address Range Sets that you use in Discovery schedules to discover configuration items (CI).

Configure **SNMP credentials** or (optionally) **SSH credentials**. Port 161 must be open for SNMP access and port 22 for SSH access.

**Network Discovery properties**

The default values are correct for most discoveries. Navigate to **Discovery Definitions > Properties** to edit the properties.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.bgp_router_disable</td>
<td>BGP router exploration disable: Disables running the <strong>SNMP – Routing</strong> probe or pattern when discovering a router running the BGP protocol. Normally such exploration IS disabled because of the huge size of BGP routing tables, and because generally such routers are only operating at the edge of large networks where further network discovery would be irrelevant. This value could be set to false if your organization only uses BGP routers as edge routers between relatively small networks (such as between buildings on a single campus).</td>
</tr>
</tbody>
</table>
| | **Type**: true | false  
| | **Default value**: true |
| glide.discovery.debug.network_discovery | Network discovery debugging: Enables extensive logging of all Network Discovery activities on the instance. |
| | **Type**: true | false  
| | **Default value**: true  
<p>| | <strong>Learn More</strong>: Discovery monitoring and issue resolution |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.discoverable.network.max.netmask.bits</td>
<td>Maximum netmask size for discoverable networks (bits): The maximum number of bits in a regular netmask for networks that are discovered by Network Discovery. A regular netmask is a netmask that can be expressed in binary as a string of ones followed by a string of zeroes. For example, 255.255.255.0 is regular and 255.255.255.64 is irregular. Regular networks are commonly expressed like: 10.0.0.0/24, which means a network address of 10.0.0.0 with a netmask of 255.255.255.0. Larger bit numbers mean networks with smaller numbers of addresses in them. For example, the network 10.128.0.128/30 has four addresses in it: one network address (10.128.0.128), one broadcast address (10.128.0.131), and two usable addresses (10.128.0.129 and 10.128.0.130). Small networks are commonly configured in network gear to provide loopback addresses or networks used strictly by point-to-point connections. Since these sorts of networks generally don't need to be discovered by Network Discovery, it would be useful to filter them out. By setting this property to a value of 1 through 32, you can limit the sizes of regular networks that are discovered. Setting it to any other value causes all networks to be discovered. Irregular networks are always discovered. The default value is 28, which means that regular networks with 8 or fewer addresses cannot be discovered.</td>
</tr>
<tr>
<td>Type</td>
<td>integer</td>
</tr>
<tr>
<td>Default value</td>
<td>28</td>
</tr>
<tr>
<td>glide.discovery.network_owner_method</td>
<td>Network router selection method: This property controls the method used to decide (during Network Discovery) which router should be selected as the router to be associated with a given IP Network. The possible values are:</td>
</tr>
<tr>
<td>First Router</td>
<td>the first router that discovers the network is associated</td>
</tr>
<tr>
<td>Last Router</td>
<td>the last router that discovers the network is associated</td>
</tr>
<tr>
<td>Most Networks</td>
<td>the router with the most attached networks is associated</td>
</tr>
<tr>
<td>Least Networks</td>
<td>the router with the least attached networks is associated</td>
</tr>
<tr>
<td>Type</td>
<td>choice list</td>
</tr>
<tr>
<td>Default value</td>
<td>Most Networks</td>
</tr>
<tr>
<td>Additional options</td>
<td>First Router, Last Router, Least Networks</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| glide.discovery.physical_interface_types  | Physical interface types: A comma-separated list of interface types that are considered physical for the purposes of network discovery. If a router (or device capable of routing) has an interface of this type, the networks connected to that interface are considered locally connected to that device. The default interface types include Ethernet, 802.11, and Token Ring types. Interface type numbers are defined in the SNMP MIB-2, specifically in OID 1.3.6.1.2.1.2.1.3.  
   - **Type**: string  
   - **Default value**: 6,117,9,71,209  

| glide.discovery.switch_interface_types     | Switch interface types: A comma-separated list of interface types that are considered. Interface type numbers are defined in the SNMP MIB-2, specifically in OID 1.3.6.1.2.1.2.1.3. Devices with any interface types that do not appear in this list are classified as routers, when they have routing. A complete list of the interface type numbers may be found on the IANA web site, in the section 'ifType definitions'.  
   - **Type**: string  
   - **Default value**: 7,8,9,26,53,62,69,71,78,115,117,209  

---

**Run network Discovery**  
Configure network Discovery from a Discovery schedule.  

Configure **SNMP credentials** or (optionally) **SSH credentials**. Port 161 must be open for SNMP access and port 22 for SSH access.  

1. Navigate to **Discovery > Discovery Schedules**.  
2. Click **New** and select **Networks** from the list in the Discover field.  
3. Select a MID Server.  
   - This field is mandatory.  
4. Complete the form, including the scheduling fields.  
5. Right-click in the header bar and select **Save** from the pop-up menu.  
   - The **Related Links** and related lists appear.  
6. Click **Network Discovery Private IPs** in Related Links to view the list of default private IP networks in the Discovery IP Ranges Related List.  
   - The default IP networks in this list are available to every Network Discovery you conduct and are sufficient for most discoveries.  
7. If your organization has additional private IP addresses, click **New** to add them.
8. Add starting routers to the schedule in the Discovery Range Sets list.
   a) Click the **Network Discovery Auto Starting Routers** link to populate the list with the starting router for each MID Server in your network.
b) Click **Edit** to add or delete routers from the list.
### Discovery Schedule - Enterprise Networks

<table>
<thead>
<tr>
<th>Name</th>
<th>Active</th>
<th>Max run time</th>
<th>Run</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Networks</td>
<td>✔️</td>
<td>Days: 00</td>
<td>Daily</td>
<td>Hours: 00, Minutes: 00</td>
</tr>
</tbody>
</table>

**Advanced**

- **Include alive**: No
- **Shazzam batch size**: 5,000
- **Shazzam cluster support**: Yes
- **Use SNMP version**: v1/v2c

**Related Links**

- Quick ranges
- Discover now
- Network Discovery private IPs
- Network Discovery auto-starting routers

**Discovery IP Ranges**

<table>
<thead>
<tr>
<th>Discovery Range Sets</th>
<th>Discovery Range Sets</th>
<th>Discovery Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Active</td>
<td>Overriding behavior</td>
</tr>
<tr>
<td>Start/End Router 10.10.10.1</td>
<td>Yes</td>
<td>False</td>
</tr>
</tbody>
</table>

**Actions or selected rows...**
9. Run Discovery manually, or through the scheduler.

_Convert IP networks into Discovery range sets_

After you conduct a network Discovery, you must convert the IP networks that were found into range sets for use in discovering other devices.

Role required: agent_admin or admin

1. Navigate to **Discovery > Output and Artifacts > IP Networks**.
2. Click **New** to add an IP network.
3. Ensure that the following field values are provided:
   - **State**: Ready
   - **Discover**: selected (true)
   - **Router**: Select a router from the list. This field must not be empty.

   If your IP Networks were created through network Discovery, then these fields are populated automatically in the IP Networks related list. However, if you entered the IP Networks manually, and you want to convert your IP Networks into range sets manually, you must edit these fields accordingly.
4. Click **Update** to return to the list of IP Networks.

5. In **Related Links**, click **Create Range Sets**.

   Use this feature to convert all the IP networks in the list into range sets.
### IP network list

The **Discovery Status** page appears, displaying the progress of the conversion. The system increments the **Started** and **Completed** count of IP networks, until all the networks are converted.

Add discovery range sets to **Discovery schedules**.  
**Add a Discovery range set to a Discovery schedule**  
After creating discovery range sets from a network discovery, add the sets to a discovery schedule. This process narrows down the range of IP addresses that Shazzam probes can scan. You can also add a behavior.

**Role required:** agent_admin or admin

1. Open a **Discovery schedule**.
2. In the **Discovery Range Sets** related list, click **Edit** to add an existing range set and use the slushbucket to add the range set.
3. On the **Discovery schedule** form, click the name of the range set under the **Range** column in the **Discovery Range Sets** related list.
4. On the **Discovery Range Set** form, add a **Discovery behavior** if necessary. You can also activate or deactivate the range set for this schedule.
5. Right-click the header, and then click **Save**.

### Cloud Discovery

Cloud Discovery provides a wizard that allows you to create and run cloud schedules in a single interface. When you create a schedule with the Discovery Manager, you select the accounts to discover, the credentials for accessing these accounts, and the MID Servers to scan the resources. You can then view the results in the Discovery Home page and track any errors that might have occurred.
How Cloud Discovery works

You can select an existing service account to discover or create one with the wizard. Select the appropriate credentials for the account and test the connection. If the connection is successful, Discovery returns the logical datacenters associated with the account, as well as any sub-accounts for selection. The cloud schedule creation process launches the necessary patterns to discover the sub-accounts and datacenters you select.

The Discovery Manager offers you the option to discover your cloud resources (virtual machines). Cloud Discovery by IP ranges finds all the VMs in a cloud service account. Just select a MID Server configured for Cloud Discovery and create the schedule. Discovery determines the IP addresses to scan, based on the IP ranges in your cloud service account. Make sure you have one or more MID Servers up and running that can access those addresses.

Note: You do not need the Cloud Management plugin to use Cloud Discovery. The Discovery plugin includes the necessary components from Cloud Management to perform Cloud Discovery.

Managing your cloud resources

The instance displays all results from a Cloud Discovery in the Discovery Home page. You can drill down into specific schedules, discovered devices, cloud resources, and errors. Sort these results for a closer look and view Discovery trends for the devices you find. Error results give you specific suggestions for fixing any problems you encounter.

Your instance can detect notifications and alerts from AWS and Azure clouds. You can configure these events to make the necessary updates to your CMDB without additional scanning. AWS or Azure event processing requires the Cloud Management plugin and is configured in the application. For more information, see Configure the Amazon AWS Config service to auto-update the CMDB and Configure the Azure Alert service to auto-update the CMDB.

AWS master accounts for AWS Organizations

An AWS organization is a collection of AWS accounts under a single account. Cloud discovery refers to AWS Organizations in the wizard as master accounts. The member accounts that belong to a master account are called sub-accounts.

Note: Cloud discovery on AWS Organizations is not fully supported in a GovCloud isolated region.

The advantages of using master accounts are:

**Easy population of sub-accounts**

After you configure the master account and supply the necessary credentials, you can test the connection to it. Discovery then detects the member accounts in that master account and displays them for selection. You can choose one or more sub-accounts to include in the Discovery of the master account.

**Discovery of sub-account resources using dynamically acquired credentials**

When you run Discovery on your cloud resources, you do not need separate credentials for each sub-account. The Cloud Discovery process handles credentials automatically by acquiring a temporary credential for each sub-account via an AWS API. You can elect to use the default
configuration or customize the MID Server to assume other roles for additional controls and security.

Cloud Discovery process

1. Install and validate a MID Server using the procedure in MID Server installation. If you do not have a MID Server running in your environment, you can use Guided Setup to quickly install and validate a MID Server. Make sure you configure Cloud Discovery MID Servers with the Cloud Management capability.

2. Open the Discovery Manager and configure the following:

   **Service account**
   You can provide the information for an existing account or add an account. If you have an AWS organization with several member accounts that belong to it, you can designate a service account as a master account, and then select any sub-accounts to discover.

   **Cloud credentials**
   Select and test your cloud credentials or add new credentials. If you are using AWS, you can configure IAM roles in an AWS instance profile that will grant temporary credentials to your master account without requiring credentials in the instance. To receive temporary AWS credentials for one or more member accounts, you can assume an AWS member role.

   **Datacenters**
   Select specific datacenters in your account to discover or configure Discovery to automatically include all current and future datacenters.

   **Virtual machine (VM) resources**
   Optionally, configure the schedule to discover VM resources by IP address. Discovery returns additional information about the VMs, including installed software and running applications.

   **Note:** For the Cloud Insights application, Cloud Discovery does not display the Discover Virtual Machines (optional) phase in the Discovery Manager tab bar.

   **Schedule**
   Create a Cloud Discovery schedule. You can run your schedule immediately from the Discovery Manager or exit the configuration and let Discovery run as scheduled.

3. View Discovery results on the Discovery Home page. You can view details for:
   - Schedules
   - Discovered Devices (VMs)
   - Cloud Resources
   - Errors

Using a MID Server through a proxy

If your MID Server must go through an unauthenticated proxy server to access cloud resources, you must modify the `agent\conf\wrapper-override.conf` file, as follows:

```
wrapper.java.additional.1=-Dhttp.proxySet=true
wrapper.java.additional.2=-Dhttp.proxyHost=<proxyHost>
```
As with any changes to the MID Server files, restart the MID Server and then test Discovery.

**Using the Discovery Manager**

Use the Discovery Manager wizard to quickly create a schedule for discovering elements of your cloud service account. Specify your provider account, the MID Servers to use, and the datacenters you want to discover. Optionally, you can schedule a deep discovery of the running applications on the virtual machines that are discovered for a selected cloud service account.

**Accessing the Discovery Manager**

To access the Discovery Manager, navigate to **Discovery > Home**. Click **Discover Cloud Resources** in the Welcome screen to begin configuring a schedule to discover your cloud resources.

**Note:** The Welcome screen contains a link to the **Discovery Quick Start**. This wizard is designed to get you up and running with a basic Discovery quickly. Use the Quick Start to evaluate Discovery or to explore new network devices. It is not intended for discovering your cloud resources.

If the system cannot detect a MID Server, you are prompted to install and configure one, using **MID Server Guided Setup**. This notice also displays if an existing MID Server is not available because it is upgrading.
Discovery Manager interface

Use the Discovery Manager to input the information you need to access your service accounts and sub-accounts, including credentials and MID Servers. Test the connections to your accounts and then create schedules to execute Discovery on the cloud resources in the datacenters you have selected.

Progress indicator

As you complete each configuration phase, tabs at the top of the Discovery Manager track your progress. A check mark indicates a completed phase. A dark underline marks the current phase. The tabs are interactive, so you can move to a phase by clicking on that tab.

Select a service account and test your credentials

The Cloud Discovery wizard begins with the account selection and testing phase. Specify your cloud provider and select the service account to discover. Discovery allows you to include one or all available sub-accounts (AWS only) on the schedule. Provide the credentials for the cloud service account and test the connection.

Before you attempt to configure Cloud Discovery, you must have at least one validated MID Server, with a status of Up, that is capable of reaching the IP ranges you intend to discover. You can install the MID Server using Guided Setup or by manually downloading and running the installer. For details, see the MID Server installation instructions for Linux and Windows. Make sure you configure MID Servers used for Cloud Discovery with the Cloud Management capability.

Role required: discovery_admin, admin

The cdud.record.watcher.timeout system property defines the maximum waiting period for these Cloud Discovery configuration processes to run:

- Service account validation
- Member account Discovery
• Discovery of logical datacenters

The default timeout value is 120,000 milliseconds, or 2 minutes.

1. In the Basic Info pane of the Discovery Manager, select the name of your cloud provider, such as AWS.
2. In the Schedule Name field, enter a unique and descriptive name for this schedule.
   The wizard displays the count of the validated and available MID Servers.
3. If no MID Servers have the Cloud Management capability, click Configure MID Servers and configure a MID Server appropriately.
   a) Navigate to MID Servers > Capabilities.
   b) Select the Cloud Management capability.
      You can also select the ALL capability, which includes Cloud Management by default.
   c) Click Edit in the MID Servers related list, and then add your MID Servers to the capability.
   d) Click Save.
      Your MID Servers are ready to communicate with your cloud service accounts.

4. In the Service Accounts pane, select the existing account for the newly named schedule.
   If you select an existing account, the system auto-completes the form with the available information. If you add an account, you must provide the account ID, credentials, and URL, if applicable. For VMware, the URL is mandatory.
   • The AWS account ID is the 12-digit account number.
   • The Azure account ID is the Subscription ID.
   • The IBM account ID is also called the Account ID.
The **GCP account ID** is the Project ID.

The **VMware account ID** is the virtual machine instance UUID.

**Note:** VMware discovery can run without a service account. To discover VMware resources, configure the schedule to run on the [IP address of the vCenter server](https://example.com).

5. Select the credentials for this cloud service account or add new credentials.
   a) To add a credential, click the plus icon (+).
   b) Select the appropriate [cloud credential type](https://example.com) from the list.
   c) Complete the form and click **Save**.

**Note:** AWS users can configure AWS to generate temporary credentials for service accounts. This capability eliminates the need to define some credentials on the instance and can be used to increase security on your sub-accounts. For details, see [Assume an AWS role for temporary cloud Discovery credentials](https://example.com).
6. Click **Test Account** to validate the account with the details provided.

Messages in the header bar notify you of the validation results. If you specified a master account and the connection test was successful, the wizard displays a selection screen for any sub-accounts associated with the master account. By default, all current sub-accounts are selected for the schedule and cannot be moved to the **Available** list.

**Note:** If the validation process does not return any sub-accounts or sub-accounts that you added recently, click **Refresh Members** or re-run the validation process. If validation times-out repeatedly, increase the timeout interval in the `cdu.record.watcher.timeout` MID Server property.
7. To select specific sub-accounts for the schedule, deactivate the auto-select feature. By default, all sub-accounts are selected. To disable the feature and to permit the selection of specific datacenters to discover, slide the toggle switch to the left.
8. Drag and drop the sub-accounts you want to exclude from the schedule from the **Selected** list to the **Available** list.

Select a cloud provider datacenter to discover

Discovery displays the list of available datacenters when you validate the service account. Specify which datacenters you want to discover with this schedule.

Role required: discovery_admin, admin

1. To select the datacenters to discover, click the **Select Datacenters** phase in the Discovery Manager tab bar.

   By default, all datacenters associated with your service account are selected for the schedule.

   **Note:** If no datacenters appear, click **Refresh Datacenters** or re-run the account validation. If validation times-out repeatedly, increase the timeout interval in the `cdu.record.watcher.timeout` MID Server property.
2. To select specific datacenters for the schedule, deactivate the auto-select feature by moving the toggle switch to the left.
3. Drag and drop the datacenters you want to discover from the **Available** list to the **Selected** list.
4. Click the next phase in the tab bar.
   Optionally, you can configure IP ranges to discover virtual machines (VM), or move to the schedule phase and complete Cloud Discovery setup.

Enable virtual machine Discovery in your cloud environment

Virtual machine (VM) Discovery is an optional configuration that performs a deep discovery of the virtual machines identified by the cloud schedule, using the IP addresses of the VMs. VM Discovery is enabled by default in the Discovery Manager wizard.
All MID Servers you intend to use for Cloud Discovery must be validated, have a status of **Up**, and be able to access the IP ranges in your cloud provider account.

**Attention:** The upgrade process from Madrid to New York deactivates existing schedules for the discovery of cloud resources. Make sure to re-activate any schedules in the Discovery Manager that might have been deactivated during an upgrade from Madrid. These schedules have a Run status of **undefined** in the list view of the Discovery Schedule (discovery_schedule) table.

Role required: admin

You do not need to configure IP ranges for the MID Servers used with Cloud Discovery. Discovery automatically selects the IP addresses of the virtual machines in the datacenters you selected in the wizard.

1. To discover the VMs that are part of the current schedule, click the **Discover Virtual Machines (optional)** phase in the Discovery Manager tab bar.
   By default, VM discovery is active.
2. Select the MD Server to use.
   The available MID Server selection types are:
   - **Auto-Select MID Server:** Discovery selects an appropriate MID Server automatically.
   - **Specific MID Cluster:** Discovery uses the named MID Server cluster.
   - **Specific MID Server:** Discovery uses the named MID Server.

3. To prevent VM Discovery from running, toggle the activation switch to the left.
   The **MID Selection Type** field disappears.
4. When you have completed this configuration, click the **Create Schedule** phase in the tab bar.

Create a Cloud Discovery schedule
The final phase in the Discovery Manager wizard creates the schedule that triggers Cloud Discovery.

Role required: admin, discovery_admin
You can also access the schedule phase of the Discovery Manager from the Discovery Schedule form. Navigate to **Discovery > Discovery Schedules** and click **Cloud Discovery** in the schedules list.

1. To configure the Cloud Discovery schedule, click **Create Schedule** in the Discovery Manager tab bar.
   
   By default, the schedule is active.

2. Configure when and how often the schedule should run and define a timeout threshold if you want to restrict Cloud Discovery to a specific duration.

3. To deactivate the schedule, toggle the switch to the left.
   
   This prevents Cloud Discovery from running.

4. Complete the Cloud Discovery configuration process by selecting one of these options:
   
   - **Finish and Run**: Runs Discovery immediately, using the current configuration. All subsequent cloud discoveries run on the schedule you provided.
   
   - **Finish**: Exits the wizard and runs Cloud Discovery on the schedule you provided.

5. Open the **Discovery Home page** and view the results of your cloud Discovery.

**Cloud Discovery account credentials**

Provide the credentials needed to access your cloud service accounts or create new credentials.

The ServiceNow® platform provides these cloud credential types:

- AWS Credentials
- Google API Credentials
- IBM Credentials
- VMware Credentials
- Azure Service Principal
**AWS Credentials**

You do not need to configure credentials for accounts that are member accounts of an AWS Organization or if IAM Instance Profiles are used. If a service account is a member account, the Discovery process automatically generates a temporary credential for the account through AWS.

**AWS Credentials form fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for the AWS credentials.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to use the credential.</td>
</tr>
<tr>
<td>Access Key ID</td>
<td>Enter the Access key ID that you generated on the AWS Management Console, such as: APIAIOSFODNN7EXAMPLE.</td>
</tr>
<tr>
<td>Secret Access Key</td>
<td>Enter the Secret access key that you generated on the AWS Management Console, such as: wPaliXUtnFEMI/K7MDENG/bPxrffYEXAMPLEKEY.</td>
</tr>
</tbody>
</table>

**Google API Credentials**

If your cloud resources are in Google Cloud Platform (GCP), create credentials that can access your GCP account.

**Google API credentials form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name of the GCP credentials to register with the instance. Ensure that this credential is easy to identify as belonging to GCP, so that it can be selected for a cloud schedule in the Discovery Manager.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to use the credential.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Determines whether this credential is available to specific MID Servers or all MID Servers.</td>
</tr>
<tr>
<td>Order</td>
<td>Order (sequence) in which the Discovery tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order.</td>
</tr>
<tr>
<td>Email</td>
<td>Email ID used to create your service account (or “profile”) in GCP.</td>
</tr>
<tr>
<td>Secret Key</td>
<td>Key provided by Google. This key must include any leading and trailing characters and must not be altered in any way.</td>
</tr>
</tbody>
</table>
IBM Credentials

If your cloud resources are in an IBM cloud, create credentials that can access your IBM account.

### IBM credentials form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential alias</td>
<td>Specifies the credentials that a Discovery schedule can use. Use credential aliases to prevent the creation of credential affinities that use inappropriate or sensitive credentials.</td>
</tr>
</tbody>
</table>

### IBM Credentials

If your cloud resources are in an IBM cloud, create credentials that can access your IBM account.

#### IBM credentials form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name of the IBM credentials to register with the instance.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order</td>
<td>Order (sequence) in which Discovery tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Softlayer API User</th>
<th>Softlayer cloud user name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Softlayer API Key</td>
<td>Softlayer API key.</td>
</tr>
<tr>
<td>Bluemix API Key</td>
<td>Bluemix API key.</td>
</tr>
</tbody>
</table>

VMware Credentials

Applications that access VMware cloud resources need access to VMware credentials. For example, the VMware credential type allows Discovery to explore VMware's vCenter running on a Windows machine to discover ESX machines, virtual machines, and resource pools. The VMware Discovery and automation API (vCenter API) now provides the globally unique serial number for computer CIs. CIM credentials are not needed to allow access to each VMware host.

**Note:** Windows credentials are not necessary for vCenter Discovery, when valid VMware credentials are used.

#### VMware credentials form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for the VMware credentials.</td>
</tr>
<tr>
<td>Active</td>
<td>Enable or disable these credentials for use.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>User name</td>
<td>Enter the user name you use for your VMware account. Avoid leading or trailing spaces in user names. A warning appears if the platform detects leading or trailing spaces in the user name. The VMware credentials must have the read-only role in vCenter.</td>
</tr>
<tr>
<td>Password</td>
<td>Enter the password for the VMware account.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Select one or more MID Servers from the list of available MID Servers. The credentials configured in this record are available to the MID Servers in this list. This field is available only when you select Specific MID servers from the Applies to field.</td>
</tr>
<tr>
<td>Order</td>
<td>Order (sequence) in which Discovery tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order.</td>
</tr>
</tbody>
</table>

Create credentials for Azure Cloud Discovery

If your cloud resources are in an Azure cloud, create credentials that can access the Azure account. This procedure requires configuration in your Azure account.

Role required: discovery_admin, sm_admin, sn_cmp.cloud_admin

Note: On the Azure portal, the Active Directory administrator or Azure administrator role is required.

A service principal for Azure cloud services is similar to a Microsoft Windows service account that enables Windows processes to communicate with each other within an Active Directory domain.

While you create the Azure service principal, you copy service principal credential values from the Azure portal into a text editor. The text file that you generate during this procedure might look something like this:
This table shows you the location in Azure where you can find the values you need for the credentials.

<table>
<thead>
<tr>
<th>Cloud Management setting</th>
<th>Location of the Azure value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tenant ID</td>
<td>Azure Active Directory &gt; Properties &gt; Directory ID</td>
</tr>
<tr>
<td>Client ID</td>
<td>Azure Active Directory &gt; App registrations &gt; Registered App.Application ID</td>
</tr>
<tr>
<td>Secret Key</td>
<td>Azure Active Directory &gt; App registrations &gt; Registered App &gt; Settings &gt; Keys (hidden)</td>
</tr>
<tr>
<td>Account ID</td>
<td>Azure Active Directory &gt; Subscriptions &gt; Subscription ID</td>
</tr>
</tbody>
</table>

1. Log in to the Azure portal and navigate to **Azure Active Directory**.
2. In the **Properties** section, click the **copy to clipboard** shortcut for the **Directory ID** field.
3. Paste the **Directory ID** value (the UUID) into the text editor and label the value **Directory ID**.
4. In the Azure portal, navigate to the **App registrations** section and click **New application registration**.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name for the application and its integration credentials. For example, ServiceNow Integration.</td>
</tr>
<tr>
<td>Application type</td>
<td>Web app / API</td>
</tr>
<tr>
<td>Sign-on URL</td>
<td>URL that will access Azure. Typically the URL of the ServiceNow instance.</td>
</tr>
</tbody>
</table>

5. Click **Save** and then select the application in the list.
6. Hover over the **Application ID** and click **Copy to clipboard**.
7. Paste the Application ID into the text editor and label it **Application ID**.
8. In the Azure portal, navigate to the **Keys** section and then specify the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key description</td>
<td>Description for the key.</td>
</tr>
<tr>
<td>Duration</td>
<td>The default is <strong>Never Expires</strong>.</td>
</tr>
</tbody>
</table>

**Note:** Your organization may apply policies to restrict key durability. Select the appropriate duration.

9. Click **Save**.
10. Paste the key value into the text editor and label the value **Application key**.
11. To enable the service principal to work with various Azure subscriptions, navigate to **Subscriptions**. To manage multiple subscriptions, you must perform the following procedure for each subscription:
    a) Paste the subscription ID into the text editor and label it **Subscription ID**.
    b) Navigate to the subscription and select **Access Control (IAM)** from the menu.
    c) Click **+ Add** at the top of the screen.
    d) Select the value **reader** from the **Role** field. Let the default value **Azure AD user, group, or application** remain as is in the **Assign access to** field.

**Note:** The **contributor** role is only required for provisioning.

    e) Select the name you created in step 4 in the **Select** field and click **Save**.
12. In the Discovery Manager, click the plus icon (⁺) and then select **Azure Service Principal** from the list.

13. Specify the following values on the Azure Service Principal form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the service principal to register with the instance. For example, Azure service principal credentials.</td>
</tr>
<tr>
<td>Authentication Method</td>
<td>Select <strong>Client secret</strong>. The <strong>Secret key</strong> field appears when you select <strong>Client secret</strong>.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> <strong>Client assertion</strong> is not supported.</td>
</tr>
</tbody>
</table>

14. Copy and paste values from the temporary text file into the remaining fields.
15. Click **Save** to create the Azure service principal.

16. Click the **Discover Subscriptions** related link to find all subscriptions for the Azure service principal. The instance creates a service account for each discovered subscription. The **Azure Subscriptions** related list displays all subscriptions for the Azure service principal.

17. Click a subscription to view the service account created for the subscription.

18. Click a Discovery status entry in the **Credential Discovery Status** list to view the Discovery log. Each time you click **Discover Subscription**, the instance generates a new Discovery status and displays it in the **Credential Discovery Status** list.

### Assume an AWS role for temporary cloud Discovery credentials

You can configure the MID Server to assume an AWS role that provides temporary cloud service account credentials. Using AWS credentials eliminates the need to manage account credentials on your instance.

AWS offers temporary credentials for the following types of service accounts:

- **Discrete account**: Standalone account, with no parent account. The `is_master_account` check box in the Discovery Manager is cleared (false).
- **Master account**: Master account that may or may not contain member accounts (sub-accounts). The `is_master_account` check box in the Discovery Manager is selected (true).
- **Member account**: Account that has a parent account. The `is_master_account` check box in the Discovery Manager is cleared (false).

Use these features to receive temporary credentials for your accounts:

- **Identity and Access Management (IAM) roles**: Provides temporary credentials granted by an AWS role for the discovery of discrete accounts and master accounts. IAM roles are defined in EC2 instance profiles. MID Servers installed on an EC2 instance can use the temporary credentials available to these roles to discover cloud resources.

- **AWS Security Token Service API**: Provides temporary credentials for any member account in a master account that does not have a permanent credential specified in the instance. The MID Server assumes a role that provides temporary credentials for a given member account, when discovering cloud resources in that member account. The ServiceNow instance includes a default configuration that provides credentials automatically for all member accounts that belong to an organization controlled by a master account. Advanced configuration allows you to customize the roles and other parameters that a MID Server can assume. You can use this capability to restrict access to certain member accounts, which enhances security.

**AWS IAM roles in EC2 instance profiles**

Amazon EC2 uses an AWS instance profile to create temporary security credentials for discovering cloud resources. These credentials are provided and managed by Identity and Access Management (IAM) roles in the profile and rotate automatically.

**Delegating permissions for a MID Server**

Use IAM roles for a MID Server installed on Amazon EC2 within the AWS cloud that is configured to discover cloud resources in "discrete" accounts. A "discrete" account is one that is not an organization - an account with no member accounts. This method of delegating permissions within the cloud eliminates the need to manage AWS account credentials on your instance. Discovery uses the credentials from AWS if an instance profile is configured in AWS for the EC2 instance.

The MID Server retrieves the security credentials from AWS that are provided by the role. The MID Server is then granted the permissions for the actions and resources defined for that role through those security credentials. These security credentials are temporary and rotate automatically. AWS generates new credentials at least five minutes before the expiration of the old credentials.

For instructions on configuring roles for an AWS instance profile, see Amazon Web Services documentation on IAM Roles for Amazon EC2.
EC2 Instance Configuration with IAM role

Configuring member accounts to use temporary credentials

To ensure that Cloud Discovery can discover member accounts with temporary credentials, configure the AWS member accounts with:

- A role named `OrganizationAccountAccessRole`.
- A trusted relationship between the `OrganizationAccountAccessRole` role and the AWS master account.
- An attached policy called `AdministratorAccess` which allows all (*) actions on all (*) AWS resources, as in this example:

```json
{
```

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Configure the MID Server for AWS IAM roles

If you configure an IAM role for an AWS instance profile, configure the MID Server to retrieve the temporary security credentials from AWS that are provided by the role.

Role required: admin

1. Navigate to Discovery > MID Servers.
2. Select a MID Server to use for AWS Cloud Discovery. This MID Server must be installed on an EC2 server within the AWS cloud.
3. Open the Configuration Parameters related list.
4. Click New.
5. In the configuration parameter form, select mid.aws.instance_profile_name in the Parameter name field.
6. Enter the IAM role name from the EC2 Instance Configuration record in the Value field.
7. Click Submit.

The new parameter is listed in the MID Server record.
Assuming member roles with an AWS API

The MID Server can call an AWS API and use the permanent credentials of an AWS master account (organization) to assume the role of one or more member accounts. By assuming the role, the MID Server receives temporary credentials for the member accounts generated by AWS for that role.

Assuming member account roles in a large AWS organization is more convenient and offers better security than using large numbers of permanent credentials for all member accounts. Temporary credentials are only acquired on behalf of a member account when there is no permanent credential specified for that member account in the Service Accounts [cmdb_ci_cloud_service_account] table.

The MID Server uses the `AssumeRole` action in the `AWS Security Token Service API` to assume a member account role. Parameters passed to this API determine what additional security restrictions are applied to the role when it accesses AWS resources.

**Default member role configuration**

By default, the MID Server is configured to assume the `OrganizationAccountAccessRole`, which grants temporary credentials to all the members of a master account. This action occurs automatically if no permanent credentials exist for the member accounts. This configuration does not apply any additional security or restrict access to any resources in member accounts.
Advanced member role configuration

You can improve security by defining additional roles that a MID Server can assume. These roles can have access to certain member accounts and determine the actions that Discovery is allowed to take on the resources in those accounts.

Create records in the Cloud Management AWS Org Assume Role Parameters module that specify the roles and restrictions that apply. Records in the Cloud Service Account AWS Org Assume Role Params (cloud_service_account_aws_org_assume_role_params) table pass their parameters to the AWS Security Token Service API, which then provides the appropriate credentials and permissions to the MID Server.

How Discovery determines which credentials to use

Cloud Discovery uses the following logic to determine which credentials to use to discover AWS cloud resources in member accounts:

1. If permanent credentials are defined for the member account in the Cloud Service Account (cmdb_ci_cloud_service_account) table, Discovery uses those credentials.
2. If no permanent credentials are defined for the member account, Discovery checks the Cloud Service Account AWS Org Assume Role Params (cloud_service_account_aws_org_assume_role_params) table for any special parameters associated with the member account. If parameters exist in that table, Discovery uses the temporary credentials acquired from specifying a role and its parameters in the AWS Security Token Service API AssumeRole action.
3. If no special parameters are associated with the member account in the (cloud_service_account_aws_org_assume_role_params) table, Discovery checks that table for parameters associated with the master account. If parameters exist that define a role for the master account, Discovery uses the temporary credentials provided by that role.
4. If no special parameters are present in the (cloud_service_account_aws_org_assume_role_params) table for either master or member accounts, Discovery uses the defaults defined for the OrganizationAccountAccessRole role.

Caching of temporary credentials

By default, temporary credentials for member accounts are cached for 60 minutes. This interval allows the horizontal Discovery process to run multiple times without generating new credentials during each Discovery.

These MID Server properties control credential caching:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| mid.aws.sts.assume_role.disable_credential_caching | Set this property to true to prevent the caching of the temporary AWS credentials.  
- Type: true | false  
- Default: false |
### Configure a custom AWS member role

Customize the AWS roles that a MID Server can assume to receive temporary credentials for member accounts. You can configure additional parameters to improve security and customize the way that the member account’s role is assumed when discovering cloud resources. To configure custom AWS member roles with the procedure, see Configure a custom AWS member role.

**Data collected for AWS and Azure Cloud Discovery**

Discovery collects information about cloud resources in AWS and Azure datacenters.

**How Discovery finds AWS and Azure resources**

Discovery uses the Cloud Management API and patterns to find cloud resources.

#### Default patterns

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon AWS Elastic Load Balancer Service</td>
<td>Retrieves AWS load balancers and populates the Load Balancer Services (cmdb_ci_lb_service) table. Application load balancers, network load balancers, and classic load balancers are supported.</td>
</tr>
<tr>
<td>Amazon AWS Relational Database Service</td>
<td>Retrieves RDS instances and populates the Cloud Database (cmdb_ci_cloud_database) table.</td>
</tr>
<tr>
<td>Amazon AWS Route53 HD</td>
<td>Resolves DNS names and aliases for the AWS cloud.</td>
</tr>
<tr>
<td>Azure Database</td>
<td>Retrieves Azure databases, including MSSQL, MySQL, Redis, and Documentum, and populates the (cmdb_ci_cloud_database) table.</td>
</tr>
<tr>
<td>Azure LoadBalancer TD</td>
<td>Retrieves Azure load balancers and populates the cmdb_ci_lb_service table. This pattern is only used by Service Mapping for top-down discovery, not by the Discovery application for horizontal discovery. <strong>Note:</strong> When using Service Mapping to find AWS cloud resources, this pattern is not used.</td>
</tr>
<tr>
<td>Azure WebSite</td>
<td>Retrieves Azure web servers and populates the Cloud WebServers (cmdb_ci_cloud_webserver) and IP address (cmdb_ci_ip_address) tables.</td>
</tr>
</tbody>
</table>

**Note:** Amazon Route 53 is supported.
### Data collected

**AWS Datacenter (cmdb_ci_aws_datacenter) and Azure Datacenter (cmdb_ci_azure_datacenter)**

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Region</td>
<td>region</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
</tbody>
</table>

**Availability Zone (cmdb_ci_availability_zone)**

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
</tbody>
</table>

**Virtual Machine Instance (cmdb_ci_vm_instance)**

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>CPUs</td>
<td>cpus</td>
</tr>
<tr>
<td>Disks</td>
<td>disks</td>
</tr>
<tr>
<td>Disks size (GB)</td>
<td>disks_size</td>
</tr>
<tr>
<td>Memory</td>
<td>memory</td>
</tr>
<tr>
<td>Network adapters</td>
<td>nics</td>
</tr>
<tr>
<td>VM Instance ID</td>
<td>vm_inst_id</td>
</tr>
</tbody>
</table>

**Compute Security Groups (cmdb_ci_compute_security_group)**

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
</tbody>
</table>

**Images (cmdb_ci_os_template)**

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
</tbody>
</table>
### Label | Field Name
---|---
Guest OS | guest_os
Root device type | root_device_type
Image source | image_source
Image type | image_type

#### Hardware Type (cmdb_ci_compute_template)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>vCPUs</td>
<td>vcpus</td>
</tr>
<tr>
<td>Memory MB</td>
<td>memory_mb</td>
</tr>
<tr>
<td>Local Storage GB</td>
<td>local_storage_gb</td>
</tr>
</tbody>
</table>

#### Storage Volume (cmdb_ci_storage_volume)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>Storage type</td>
<td>storage_type</td>
</tr>
<tr>
<td>Size</td>
<td>size</td>
</tr>
</tbody>
</table>

#### Cloud Networks (cmdb_ci_network) and VMware vCenter Network (cmdb_ci_vcenter_network)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>State*</td>
<td>state</td>
</tr>
<tr>
<td>CIDR*</td>
<td>cidr</td>
</tr>
</tbody>
</table>

*Not found on VMware vCenter networks.

#### Cloud Subnets (cmdb_ci_cloud_subnet)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Status</td>
<td>status</td>
</tr>
<tr>
<td>CIDR</td>
<td>cidr</td>
</tr>
</tbody>
</table>
Cloud Management Network Interfaces (cmdb_ci_nic)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Netmask</td>
<td>netmask</td>
</tr>
<tr>
<td>MAC Address</td>
<td>mac_address</td>
</tr>
<tr>
<td>MAC Manufacturer</td>
<td>mac_manufacturer</td>
</tr>
<tr>
<td>Status</td>
<td>install_status</td>
</tr>
</tbody>
</table>

Cloud Load Balancers (cmdb_ci_cloud_load_balancer)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
</tbody>
</table>

Azure Deployments (cmdb_ci_azure_deployment)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Provisioning state</td>
<td>provisioning_state</td>
</tr>
</tbody>
</table>

Note: This table is only available for Discovery when the Cloud Management plugin is activated.

Resource Groups (cmdb_ci_resource_group)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
</tbody>
</table>

Public IP Addresses (cmdb_ci_cloud_public_ipaddress)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>Public IP address</td>
<td>public_ip_address</td>
</tr>
<tr>
<td>Public DNS</td>
<td>public_dns</td>
</tr>
</tbody>
</table>
### Storage Accounts (cmdb_ci_cloud_storage_account)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
</tr>
<tr>
<td>Sku Name</td>
<td>sku_name</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
</tbody>
</table>

### DNS Alias (cmdb_ci_dns_alias) and DNS name (cmdb_ci_dns_name)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS Alias (cmdb_ci_dns_alias)</td>
<td>name</td>
</tr>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Category</td>
<td>category</td>
</tr>
<tr>
<td>Status</td>
<td>status</td>
</tr>
<tr>
<td>DNS name (cmdb_ci_dns_name)</td>
<td>name</td>
</tr>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
</tbody>
</table>

### Cloud Databases (cmdb_ci_cloud_database)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>The name of the database that you created in AWS or Azure.</td>
</tr>
<tr>
<td>Object ID</td>
<td>object_id</td>
<td>This is also the name of the database.</td>
</tr>
<tr>
<td>Type</td>
<td>Type</td>
<td>The type of database you created.</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
<td>The FQDN that AWS or Azure assigned to your database. An example format for AWS is as follows: database-name.{random-number}. {datacenter}.rds.amazonaws.com</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
<td>The state of the database: whether it is Available or Terminated.</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
<td>The TCP port that the database communicates through.</td>
</tr>
<tr>
<td>Category</td>
<td>category</td>
<td>The instance class of the database, for example: db.t2.micro.</td>
</tr>
</tbody>
</table>

**Note:** You might need to configure the form layout to see all of these fields if they are not visible. Right-click the form header and select **Configure > Form Layout**, and then move the fields you want to see to the **Selected** column.
Cloud WebServer (cmdb_ci_cloud_webserver)

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Install status</td>
<td>install_status</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>Operational status</td>
<td>operational_status</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
</tbody>
</table>

Relationships between virtual machines, datacenters, and other CIs

<table>
<thead>
<tr>
<th>Class</th>
<th>Relationship</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Machine Instance (cmdb_ci_vm_instance)</td>
<td>Hosted on</td>
<td>AWS Datacenter (cmdb_ci_aws_datacenter)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Azure Datacenter (cmdb_ci_azure_datacenter)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Virtualizes</td>
<td></td>
<td>Computer (cmdb_ci_computer)</td>
</tr>
<tr>
<td>Logical Datacenter (cmdb_ci_logical_datacenter)</td>
<td>Contains</td>
<td>Resource Group (cmdb_ci_resource_group)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Public IP Address (cmdb_ci_cloud_public_ip_address)</td>
</tr>
<tr>
<td></td>
<td>Hosts</td>
<td>Cloud Service Account (cmdb_ci_cloud_service_account)</td>
</tr>
<tr>
<td></td>
<td>Hosted on</td>
<td>Storage Account (cmdb_ci_cloud_storage_account)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Availability Zone (cmdb_ci_availability_zone)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Host Cluster (cmdb_ci_host_cluster)</td>
</tr>
<tr>
<td></td>
<td>Hosts</td>
<td>OS Template (cmdb_ci_os_template)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Compute Template (cmdb_ci_compute_template)</td>
</tr>
<tr>
<td>Class</td>
<td>Relationship</td>
<td>Class</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>Azure Datacenter (cmdb_ci_azure_datacenter)</td>
<td>Hosted on</td>
<td>Cloud Management Network Interfaces (cmdb_ci_nic)</td>
</tr>
<tr>
<td>Cloud DataBase (cmdb_ci_cloud_database)</td>
<td>Contains</td>
<td>Availability Set (cmdb_ci_availability_set)</td>
</tr>
<tr>
<td></td>
<td>Owns</td>
<td>IP Address (cmdb_ci_ip_address)</td>
</tr>
<tr>
<td></td>
<td>Hosted on</td>
<td>AWS Datacenter (cmdb_ci_aws_datacenter)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Azure Datacenter (cmdb_ci_azure_datacenter)</td>
</tr>
<tr>
<td></td>
<td>Hosted on</td>
<td>Cloud Service Account (cmdb_ci_cloud_service_account)</td>
</tr>
</tbody>
</table>

**AWS Config and Azure Alerts services**

If you configured the *AWS Config* or *Azure Alert* services, the instance can receive notifications when changes to cloud resources occur. Discovery can then take action and make updates.

**AWS config**

The instance can detect an AWS config notification with message type `ConfigurationItemChangeNotification` for these resource types:

- AWS::EC2::Instance
- AWS::EC2::VPC
- AWS::EC2::Subnet
- AWS::EC2::Volume

Discovery can then make updates to records in the Response Mappings (sn_cmp_response_mapping) table that have Cloud Event in the *Datasource* field.

**Azure alerts**

The instance can detect alerts for the following resource types:

- Microsoft.Compute/virtualMachines
- Microsoft.Network/networkSecurityGroups
- Microsoft.Resources/deployments
- Microsoft.Storage/storageAccounts
- Microsoft.Compute/availabilitySets
- Microsoft.Network/networkInterfaces
- Microsoft.Network/publicIPAddresses
- Microsoft.Network/loadBalancers
- Microsoft.Network/virtualNetworks

Discovery can then run on the specific cloud resource to retrieve the changes and update the CMDB.

*Data collected for GCP Cloud Discovery*

Discovery collects information about cloud resources in Google Cloud Platform (GCP).
# GCP data collected

<table>
<thead>
<tr>
<th>Table</th>
<th>ServiceNow Field</th>
<th>GCP Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region (cmdb_ci_google_datacenter)</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>region</td>
<td>items.name</td>
</tr>
<tr>
<td>Virtual Server (cmdb_ci_vm_instance)</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>state</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>cpus</td>
<td>Not mapped</td>
</tr>
<tr>
<td></td>
<td>memory</td>
<td>Not mapped</td>
</tr>
<tr>
<td></td>
<td>disks</td>
<td>disks (array size)</td>
</tr>
<tr>
<td></td>
<td>disks_size</td>
<td>Not mapped</td>
</tr>
<tr>
<td></td>
<td>nics</td>
<td>networkInterfaces (array size)</td>
</tr>
<tr>
<td></td>
<td>termination_protection</td>
<td>deletionProtection</td>
</tr>
<tr>
<td></td>
<td>Not mapped</td>
<td>canIpForward</td>
</tr>
<tr>
<td></td>
<td>Not mapped</td>
<td>selfLink</td>
</tr>
<tr>
<td></td>
<td>sys_created_on</td>
<td>creationTimestamp</td>
</tr>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td></td>
<td>state</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>size</td>
<td>sizeGb</td>
</tr>
<tr>
<td></td>
<td>storage_type</td>
<td>storage_type</td>
</tr>
<tr>
<td>Cloud Public IP Address (cmdb_ci_cloud_public_ipaddress)</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>public_ip_address</td>
<td>address</td>
</tr>
<tr>
<td></td>
<td>state</td>
<td>status</td>
</tr>
<tr>
<td>Availability Zone (cmdb_ci_availability_zone)</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td></td>
<td>state</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td>status</td>
</tr>
<tr>
<td>Hardware Template (cmdb_ci_compute_template)</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>Table</td>
<td>ServiceNow Field</td>
<td>GCP Field</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td></td>
<td>vcpus</td>
<td>guestCpus</td>
</tr>
<tr>
<td></td>
<td>memory_mb</td>
<td>memoryMb</td>
</tr>
<tr>
<td></td>
<td>local_storage_gb</td>
<td>imageSpaceGb</td>
</tr>
<tr>
<td>Instance Template</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td>(cmdb_ci_instance_template)</td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td>Storage Volume Snapshot</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td>(cmdb_ci_storage_vol_snapshot)</td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>state</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>volume_name</td>
<td>sourceDisk</td>
</tr>
<tr>
<td></td>
<td>capacity</td>
<td>storageBytes</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>storageBytesStatus</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>selfLink</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>labelFingerprint</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>diskSizeGb</td>
</tr>
<tr>
<td>Disk Type (cmdb_ci_disk_type)</td>
<td>object_id</td>
<td>zone:name</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name@zone</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td></td>
<td>valid_disk_size</td>
<td>validDiskSize</td>
</tr>
<tr>
<td></td>
<td>default_disk_size_gb</td>
<td>defaultDiskSizeGb</td>
</tr>
<tr>
<td>Subnet (cmdb_ci_cloud_subnet)</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>cidr</td>
<td>ipCidrRange</td>
</tr>
<tr>
<td></td>
<td>gateway</td>
<td>gatewayAddress</td>
</tr>
<tr>
<td>Image (cmdb_ci_os_template)</td>
<td>object_id</td>
<td>items.id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>image_type</td>
<td>sourceType</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>status</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>archiveSizeBytes</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>diskSizeGb</td>
</tr>
<tr>
<td></td>
<td>image_source</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• sourceDisk</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• sourceDiskId</td>
<td></td>
</tr>
<tr>
<td>Load Balancer (cmdb_ci_cloud_loadBalancer)</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>defaultService</td>
</tr>
<tr>
<td>Table</td>
<td>ServiceNow Field</td>
<td>GCP Field</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>Load Balancer Service</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td>(cmdb_ci_lb_service)</td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>pool</td>
<td>backends</td>
</tr>
<tr>
<td></td>
<td>port</td>
<td>port</td>
</tr>
<tr>
<td></td>
<td>listener_protocol</td>
<td>protocol</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>portName</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>loadBalancingScheme</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>sessionAffinity</td>
</tr>
<tr>
<td></td>
<td>No mapping</td>
<td>enableCDN</td>
</tr>
<tr>
<td></td>
<td>service_type</td>
<td>Backend/Frontend</td>
</tr>
<tr>
<td>Load Balancer Health Service</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td>(cmdb_ci_lb_health_service)</td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>monitor_type</td>
<td>type (tcp/http/https)</td>
</tr>
<tr>
<td></td>
<td>request_path</td>
<td>requestPath</td>
</tr>
<tr>
<td></td>
<td>port</td>
<td>port</td>
</tr>
<tr>
<td></td>
<td>unhealthy_threshold</td>
<td>unhealthyThreshold</td>
</tr>
<tr>
<td></td>
<td>healthy_threshold</td>
<td>healthyThreshold</td>
</tr>
<tr>
<td></td>
<td>timeout_sec</td>
<td>timeoutSec</td>
</tr>
<tr>
<td></td>
<td>check_interval_sec</td>
<td>checkIntervalSec</td>
</tr>
<tr>
<td>Load Balancer Pool Member</td>
<td>object_id</td>
<td>Instance ID from GCP</td>
</tr>
<tr>
<td>(cmdb_ci_lb_pool_member)</td>
<td>name</td>
<td>instance.substring after the last'/'</td>
</tr>
<tr>
<td></td>
<td>Operational status</td>
<td>status</td>
</tr>
<tr>
<td>Load Balancer Pool</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td>(cmdb_ci_lb_pool)</td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>Network (cmdb_ci_network)</td>
<td>state</td>
<td>No mapping</td>
</tr>
<tr>
<td></td>
<td>cidr</td>
<td>IPv4Range</td>
</tr>
<tr>
<td></td>
<td>is_external</td>
<td>No mapping</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>object_id</td>
<td>items_id</td>
</tr>
<tr>
<td></td>
<td>description</td>
<td>description</td>
</tr>
<tr>
<td>Network ACL (cmdb_ci_network_acl)</td>
<td>object_id</td>
<td>id</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td>description</td>
</tr>
<tr>
<td>Network ACL Rule (cmdb_ci_network_acl_rule)</td>
<td>is_outbound</td>
<td>(True/False)</td>
</tr>
<tr>
<td></td>
<td>dest_from_port</td>
<td>No mapping</td>
</tr>
<tr>
<td></td>
<td>dest_to_port</td>
<td>No mapping</td>
</tr>
<tr>
<td></td>
<td>src_from_port</td>
<td>No mapping</td>
</tr>
</tbody>
</table>
### Data collected for IBM Cloud Discovery
Discovery collects information about cloud resources in the IBM Cloud and creates relationships between the CIs it finds.

### Load balancers

#### Data collected for load balancers

<table>
<thead>
<tr>
<th>Table</th>
<th>Field Name / Description</th>
</tr>
</thead>
</table>
| Cloud Load Balancer (cmdb_ci_cloud_load_balancer) | **object_id**: The load balancer’s unique identifier.  
**name**: Name of the load balancer.  
**fqdn**: Address (Host name) of the load balancer. This is the fully qualified domain name of the load balancer.  
**dns_name**: Address (Host name) of the load balancer.  
**short_description**: Description of the load balancer.  
**operational_status**: The operational status of the load balancer, either ONLINE or OFFLINE.  
**location**: References the load balancer’s location with type.  
**state**: Provisioning status of the load balancer, either Available or Terminated. |
| Load Balancer Service (cmdb_ci_lb_service) | **object_id**: Unique identifier of the load balancer service.  
**name**: Name created by the pattern that contains the following:  
- Load balancer name  
- Protocol  
- Port  
- Resource type: -listener  
**port**: Listener’s protocol port number.  
**front_end_port**: Listener’s protocol port number.  
**listener_protocol**: Listener’s protocol, either TCP, HTTP, or HTTPS.  
**operational_status**: The provisioning status of the listener. |
<p>| Load Balancer Pool (cmdb_ci_lb_pool) | <strong>object_id</strong>: Unique identifier of the load balancer pool. |</p>
<table>
<thead>
<tr>
<th>Table</th>
<th>Field Name / Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>name</strong>: Name created by the pattern that contains the following:</td>
</tr>
<tr>
<td></td>
<td>· Load balancer name</td>
</tr>
<tr>
<td></td>
<td>· Protocol</td>
</tr>
<tr>
<td></td>
<td>· Port</td>
</tr>
<tr>
<td></td>
<td>· Resource type: -pool</td>
</tr>
<tr>
<td></td>
<td><strong>port</strong>: Protocol port of the back end.</td>
</tr>
<tr>
<td></td>
<td><strong>pool_status</strong>: Provisioning status of a load balancer pool.</td>
</tr>
<tr>
<td></td>
<td><strong>load_balancing_method</strong>: Method used to load balance the pool.</td>
</tr>
<tr>
<td>Load Balancer Pool Member</td>
<td><strong>object-id</strong>: Unique identifier of the pool member, created from the ID of the member and the listener ID.</td>
</tr>
<tr>
<td>(cmdb_ci_lb_pool_member)</td>
<td><strong>name</strong>: Name created by the pattern that contains the following:</td>
</tr>
<tr>
<td></td>
<td>· Load balancer name</td>
</tr>
<tr>
<td></td>
<td>· Protocol</td>
</tr>
<tr>
<td></td>
<td>· Port</td>
</tr>
<tr>
<td></td>
<td>· Resource type: -member</td>
</tr>
<tr>
<td></td>
<td>· IP address of the pool member</td>
</tr>
<tr>
<td></td>
<td><strong>ip_address</strong>: IP address of a load balancer member.</td>
</tr>
<tr>
<td></td>
<td><strong>server_id</strong>: ID of the member in the IBM Cloud.</td>
</tr>
<tr>
<td></td>
<td><strong>load_balancer</strong>: Name of the related load balancer.</td>
</tr>
<tr>
<td>Load Balancer Health Service</td>
<td><strong>object_id</strong>: Unique identifier of the healthservice check.</td>
</tr>
<tr>
<td>(cmdb_ci_lb_health_service)</td>
<td><strong>name</strong>: Name created by the pattern that contains the following:</td>
</tr>
<tr>
<td></td>
<td>· Load balancer name</td>
</tr>
<tr>
<td></td>
<td>· Protocol</td>
</tr>
<tr>
<td></td>
<td>· Resource type: Health Check</td>
</tr>
<tr>
<td></td>
<td>· Resource ID</td>
</tr>
<tr>
<td></td>
<td><strong>monitor_type</strong>: Type of health check. Valid values are TCP, HTTP, or HTTPS.</td>
</tr>
<tr>
<td></td>
<td><strong>timeout_sec</strong>: Timeout in seconds to wait for the health check response.</td>
</tr>
<tr>
<td></td>
<td><strong>check_interval_sec</strong>: Interval in seconds to perform the health check.</td>
</tr>
</tbody>
</table>

**Relationships created for load balancers**

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_cloud_load_balancer</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_lb_health_service</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
## Organizations and spaces

### Data collected for organizations and spaces

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Cloud Organizations (cmdb_ci_cloud_org)</td>
<td></td>
</tr>
</tbody>
</table>
  - **guid**: The organization's unique identifier.  
  - **url**: The organization's unique URL.  
  - **billing_enabled**: Indicates if the organization's billing method is enabled or disabled.  
  - **name**: Unique name of the cloud organization.  
  - **quota_definition_url**: Resources that are available to the organization.  
  - **spaces_url**: Unique URL of the space assigned to the organization.  
  - **operational_status**: Operational status of the organization. |
| IBM Cloud Space (cmdb_ci_cloud_space) |  
  - **object_id**: Unique identifier (GUID) of the space.  
  - **name**: Unique name of the space.  
  - **url**: Unique URL of the space.  
  - **allow_ssh**: The IP address of a storage resource. |

### Relationships created for organizations and spaces

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_cloud_load_balancer</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_lb_service</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_lb_pool</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_lb_pool_member</td>
</tr>
</tbody>
</table>
SSH Keys

Data collected for SSH keys

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
</table>
| Cloud Key Pairs (cmdb_ci_cloud_key_pair) | - **object_id**: ID of the SSH key record.  
- **name**: Descriptive name used to identify the SSH key.
- **key**: The SSH key.
- **finger_print**: Short sequence of bytes used to authenticate or lookup a longer SSH key. This is generated automatically when adding or modifying the SSH key. This property is read only. Changes made are ignored.
- **created_date**: Date the SSH key was added. This property is read only. Changes made are ignored.
- **modify_date**: Date the SSH key was last modified. This property is read only. Changes made are ignored.
- **short_description**: Brief note about the SSH key. |

Relationships created for SSH keys

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contained by::Contains</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_cloud_key_pair</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_cloud_key_pair</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
</tbody>
</table>
IBM Cloud storage

Data collected for IBM Cloud storage

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>* object_id: Unique identifier of the storage account.</td>
</tr>
<tr>
<td></td>
<td>* size: Capacity of the storage account, measured in gigabytes.</td>
</tr>
<tr>
<td></td>
<td>* size_bytes: Capacity of the storage account, measured in bytes.</td>
</tr>
<tr>
<td></td>
<td>* name: Username used to access a non-EVault Storage volume. This username is used to register the EVault server agent with the vault backup system.</td>
</tr>
<tr>
<td></td>
<td>* label: Network resource name for the storage.</td>
</tr>
<tr>
<td></td>
<td>* storage_type: Description of the storage object.</td>
</tr>
<tr>
<td></td>
<td>* fqdn: IP address of the storage resource.</td>
</tr>
<tr>
<td></td>
<td>* device: Unique identification number of the guest associated with the storage volume.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Storage Volume Snapshot (cmdb_ci_storage_vol_snapshot)</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>* object_id: Unique identifier of the storage account.</td>
</tr>
<tr>
<td></td>
<td>* name: Name of the storage's network resource.</td>
</tr>
<tr>
<td></td>
<td>* capacity: The volume's configured snapshot space size.</td>
</tr>
<tr>
<td></td>
<td>* fqdn: IP address of the storage resource.</td>
</tr>
<tr>
<td></td>
<td>* location: Unique availability zone where the storage volume is located.</td>
</tr>
</tbody>
</table>

Relationships created for IBM Cloud storage

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Provisioned::Provisioned From</td>
<td>cmdb_ci_storage_vol_snapshot</td>
</tr>
<tr>
<td>cmdb_ci_storage_vol_snapshot</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_storage_vol_snapshot</td>
<td>Located in Zone::Zone contains</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_storage_volume</td>
</tr>
</tbody>
</table>
## IBM Cloud virtual servers

### Data collected for IBM Cloud virtual servers

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Machine Instance</td>
<td><strong>name</strong>: Name of the virtual machine.</td>
</tr>
<tr>
<td>(cmdb_ci_vm_instance)</td>
<td><strong>object_id</strong>: ID of the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>dns_domain</strong>: Domain of the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>fqdn</strong>: IP address of the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>short_description</strong>: Notes attached to the virtual machine on the IBM console.</td>
</tr>
<tr>
<td></td>
<td><strong>cpus</strong>: Number of CPUs on the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>memory</strong>: Amount of memory on the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>disks</strong>: Number of disks used by the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>state</strong>: The running state of the virtual machine. The choices are:</td>
</tr>
<tr>
<td></td>
<td>- Running</td>
</tr>
<tr>
<td></td>
<td>- Halted</td>
</tr>
<tr>
<td></td>
<td><strong>placement_group_id</strong>: Name of the IBM placement group containing the virtual machine.</td>
</tr>
<tr>
<td></td>
<td><strong>nics</strong>: Number of network interfaces on the virtual machine.</td>
</tr>
<tr>
<td>Cloud Mgmt Network Interface</td>
<td><strong>name</strong>: Name and port of the NIC, in the format <code>&lt;name&gt;&lt;port&gt;</code>.</td>
</tr>
<tr>
<td>(cmdb_ci_nic)</td>
<td><strong>ip_address</strong>: IP address configured on the NIC.</td>
</tr>
<tr>
<td></td>
<td><strong>mac_address</strong>: MAC address of the NIC.</td>
</tr>
<tr>
<td></td>
<td><strong>netmask</strong>: Netmask of the NIC.</td>
</tr>
<tr>
<td></td>
<td><strong>object_id</strong>: ID of the NIC.</td>
</tr>
<tr>
<td></td>
<td><strong>status</strong>: State of the NIC - AVAILABLE.</td>
</tr>
<tr>
<td>Image (cmdb_ci_os_template)</td>
<td><strong>name</strong>: Long description of the template. For example: CentOS 7.0-64 Minimal for VSI.</td>
</tr>
<tr>
<td></td>
<td><strong>version</strong>: Operating system version. For example, 7.0-64 Minimal for VSI.</td>
</tr>
<tr>
<td></td>
<td><strong>object_id</strong>: ID of the image.</td>
</tr>
<tr>
<td></td>
<td><strong>guest_os</strong>: Operating system of the virtual machine, such as Windows, Ubuntu, or CentOS.</td>
</tr>
<tr>
<td>Cloud Key Pair (cmdb_ci_cloud_key_pair)</td>
<td><strong>name</strong>: The key label.</td>
</tr>
<tr>
<td></td>
<td><strong>object_id</strong>: ID of the key.</td>
</tr>
<tr>
<td>Compute Security Group (cmdb_ci_compute_security_group)</td>
<td><strong>name</strong>: Security group name.</td>
</tr>
<tr>
<td></td>
<td><strong>object_id</strong>:</td>
</tr>
<tr>
<td></td>
<td><strong>short_description</strong>: Security group notes.</td>
</tr>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td><strong>object_id</strong>: The storage volume ID.</td>
</tr>
</tbody>
</table>
Relationships created for IBM Cloud virtual servers

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_vm_instance</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_vm_instance</td>
<td>Located in::Houses</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_nic</td>
<td>Contained by::Contains</td>
<td>cmdb_ci_vm_instance</td>
</tr>
<tr>
<td>cmdb_ci_os_template</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_os_template</td>
<td>Provisioned by::Provisions</td>
<td>cmdb_ci_vm_instance</td>
</tr>
<tr>
<td>cmdb_ci_cloud_key_pair</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_cloud_key_pair</td>
<td>Used by::Uses</td>
<td>cmdb_ci_vm_instance</td>
</tr>
<tr>
<td>cmdb_ci_compute_security_group</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_compute_security_group</td>
<td>Implement End Point From::Implement End Point To</td>
<td>cmdb_ci_vm_instance</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Implement End Point From::Implement End Point To</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

IBM Cloud service account

Data collected for IBM Cloud service account

<table>
<thead>
<tr>
<th>Table</th>
<th>Field</th>
</tr>
</thead>
</table>
| Cloud Service Account (cmdb_ci_cloud_service_account) | • name: Display name of the cloud service account.  
                                                      • account_id: Service account ID.  
                                                      • object_id: Cloud ID of the service account. This is the same as the account_id.  
                                                      • datacenter_type: This is cmdb_ci_ibm_datacenter.  
                                                      • discovery_credentials: The sys_id of the credentials.  |
| IBM Datacenter (cmdb_ci_ibm_datacenter) | • name: Display name of the cloud service account.  
                                                      • account_id: Service account ID.  
                                                      • object_id: Cloud ID of the service account. This is the same as the account_id.  
                                                      • datacenter_type: This is cmdb_ci_ibm_datacenter.  |

Relationships created for IBM Cloud service accounts

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_cloud_service_account</td>
<td>Contained by::Contains</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_cloud_service_account</td>
<td>Contained by::Contains</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
</tbody>
</table>
Data collected for VMware Cloud Discovery
Discovery collects information about VMware resources in your cloud service accounts.

Distributed Resource Scheduler (DRS) discovery

VMware Discovery collects these DRS settings and saves them to the DRS VM Config (cmdb_ci_drs_vm_config) table:

- DRS settings of vCenter clusters.
- DRS settings of VMs configured to override cluster behavior.

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>cluster</td>
</tr>
<tr>
<td>DRS behavior</td>
<td>drs_behavior</td>
</tr>
<tr>
<td>DRS enabled</td>
<td>drs_enabled</td>
</tr>
<tr>
<td>Virtual machine</td>
<td>virtual_machine</td>
</tr>
</tbody>
</table>

VMware tags

VMware Cloud Discovery finds VMware tags from vCenter and saves them to the Key Value (cmdb_key_value) table.

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration item (configuration_item)</td>
<td>Referenced VM with tag attached in vCenter.</td>
</tr>
<tr>
<td>Key (key)</td>
<td>Category name of the tag applied to the resource in vCenter.</td>
</tr>
<tr>
<td>Tag (tag)</td>
<td>Source in vCenter from which tags are fetched. In this case, the value is always Tags.</td>
</tr>
<tr>
<td>Value (value)</td>
<td>Tag name that is applied to the resource in vCenter.</td>
</tr>
</tbody>
</table>
Serverless Discovery

Discovery can find applications on host machines without the need to discover the host first. This type of Discovery is referred to as serverless Discovery.

Serverless Discovery relies on infrastructure patterns to explore CIs on a host. This kind of Discovery skips the scanning and classification phases of Discovery. There are two types of serverless Discovery: standard and host-based. Both types require an infrastructure pattern.

Standard serverless Discovery

Standard serverless Discovery finds remote application CIs that reside on an unknown host. This type of serverless Discovery uses protocols, such as REST or HTTP, to find the CIs. The IP address of the host machine is not required. You define input parameters in an infrastructure pattern to connect to a remote resource. Use this type of Discovery when you want to discover web-based applications.

Host-based serverless Discovery

Host-based serverless Discovery finds CIs that reside on a host machine that you already discovered. The host acts as a proxy through which you find other CIs. You do not actually run Discovery on the host. Use this type of Discovery when:

- You do not know the IP address of the CIs you want to discover, or the CIs do not actually have IP addresses.
- The CIs reside on an already discovered host CI.

For this type of Discovery, you also need an infrastructure pattern, but you do not need to define input parameters like you do for standard serverless Discovery. The pattern can specify operations, such as a WMI Query, which executes queries on a host. After Discovery connects to the host, it uses the credentials of the host to perform the Discovery of CIs you are exploring in the pattern. You do not need to specify additional credentials. When you run debug mode on the pattern, you can see the details under Temporary Variables on the pattern. You can then create pattern operations that make API calls from the host to the CIs you are trying to discover.

Serverless Discovery schedules

The infrastructure patterns that serverless Discovery uses are triggered from a Discovery schedule, rather than from a classifier. Serverless Discovery ignores classifiers. Use a serverless schedule type for both types of serverless Discovery. The serverless schedule allows you to specify:

- Values for input parameters in the pattern if you are using standard serverless Discovery. See Create a schedule for standard serverless Discovery for instructions.
- A host CI if you are using proxy-host based Discovery. See Create a schedule for host-based serverless Discovery for more information.

Create a pattern and a schedule for standard serverless Discovery

To find application CIs that reside on an unknown host, create an infrastructure pattern for standard serverless discovery. After creating the pattern, create a Discovery schedule that triggers the pattern.

Role required: cloud_admin, sm_admin, or admin
In the pattern, you must specify input parameters to explore the CI.

1. Navigate to Pattern Designer > Discovery Patterns.
2. Click New.
3. Specify an Infrastructure type pattern with the correct CI type that applies to the CIs you want to discover. See Create or modify patterns for an explanation of all form fields.
4. Create an identification section and save the pattern.
5. Click the Input Parameters tab, and then click New.
6. Enter the parameters that are required to reach the serverless CIs. For example, you could enter url to reach a remote resource via an HTTP request.

Note: You specify the actual URL in a serverless discovery schedule.

7. Save the pattern.
8. Click the Basic tab, and then click the identification section name to open the pattern designer.
9. Create the steps you need to explore serverless CIs. For example, create an Http Get Call operation to reach a remote resource through a URL. In this case, you would specify the $url parameter in the URL field.
10. To test the pattern, click Debug Mode and fill in values for the input parameters. You do not need to enter a value for Proxy Host.

In this example, we are connecting to a website that serves up an XML file. The XML file parsing method is already defined in the Http Get Call operation. To test the pattern operation, click Debug Mode, enter the URL, and click Connect.

Example debugging of an HTTP Get Call

11. After you connect and run the operation, the Output field displays the content.
After you create a pattern for standard serverless discovery, create a Discovery schedule and specify the pattern. Discovery triggers the pattern from the schedule rather than from a classifier.

Ensure you have basic authentication credentials for HTTP calls in the pattern.

Role required: cloud_admin, sm_admin, or admin

You can reuse this schedule to run the same pattern using different parameters.

1. Navigate to Discovery > Schedules.
2. Click New.
3. Select Serverless in the Discovery field. See Schedule a horizontal discovery for explanation of other fields.
4. Right-click the form header and select Save.
5. In the Serverless Execution Pattern related list, click New.
6. Fill out the form fields (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive name.</td>
</tr>
<tr>
<td>Pattern</td>
<td>Select the pattern you created for standard serverless discovery or for proxy host-based discovery.</td>
</tr>
<tr>
<td>Proxy Host</td>
<td>Leave this field empty. This field applies to the host proxy-based scenario.</td>
</tr>
<tr>
<td>Active</td>
<td>Keep this check box selected to use the pattern execution definition.</td>
</tr>
</tbody>
</table>

7. Right-click the form header and select Save.
8. In the Discovery Pattern Launcher Parameters related list, add the parameters required by the pattern and provide a value for each one in the Volume column.
9. Click Submit.
10. Define as many serverless execution pattern records as needed. You can launch the same pattern, but they should have different parameters.
11. Execute discovery by clicking Discover Now or wait until the schedule triggers the discovery.

Check the Discovery status to see the Discovery log, the devices list, which holds the discovered CIs, and the ECC queue records. Note that the Shazzam probe did not launch and Discovery did not trigger any classification probes.

Create a pattern and a schedule for host-based serverless Discovery
To find application CIs that reside on a known host in your CMDB, create an infrastructure pattern for host-based serverless discovery. After creating the pattern, create a Discovery schedule that triggers the pattern.

Role required: cloud_admin, sm_admin, or admin

1. Navigate to Pattern Designer > Discovery Patterns.
2. Click New.
3. Specify an Infrastructure type pattern with the correct CI type that applies to the CIs you want to discover. See Create or modify patterns for an explanation of all form fields.
4. Create an identification section and save the pattern.
5. Add operations to the pattern to discover the CI that you can access via the host.
6. To test the pattern, click Debug Mode and enter the IP address of the host.

Create a schedule for host-based serverless Discovery
Discovery triggers the pattern for host-based, serverless Discovery from the schedule rather than from a classifier.

Ensure you have credentials for the host device that acts as the proxy.

Role required: cloud_admin, sm_admin, or admin

1. Navigate to **Discovery > Schedules**.
2. Click **New**.
3. Select the **Serverless** in the Discovery field. See *Schedule a horizontal discovery* for explanation of other fields.
4. Right-click the form header and select **Save**.
5. In the **Serverless Execution Pattern** related list, click **New**.
6. Fill out the form fields (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive name.</td>
</tr>
<tr>
<td>Pattern</td>
<td>Select the pattern you created for host-based discovery.</td>
</tr>
<tr>
<td>Proxy Host</td>
<td>If you are using host-based discovery, specify an existing CMDB CI. Discovery uses the credentials of this CI when exploring the CIs on the host.</td>
</tr>
<tr>
<td>Active</td>
<td>Keep this check box selected to use the pattern execution definition.</td>
</tr>
</tbody>
</table>

7. Right-click the form header and select **Save**.
8. Define as many serverless execution pattern records as needed. You can launch the same pattern, but they should have different parameters.

**Note**: You do not need to create entries in the **Discovery Pattern Launcher Parameters** related list.

9. Execute discovery by clicking **Discover Now** or wait until the schedule triggers the discovery.

Check the Discovery status to see the Discovery log, the devices list, which holds the discovered CIs, and the ECC queue records. Note that the Shazzam probe did not launch and Discovery did not trigger any classification probes.

**Discovery configuration**

Configure the elements that Discovery requires to investigate your network, such as credentials, schedules, and IP addresses.

**Configuration overview**

Discovery can run on a regular, configurable schedule, or can be launched manually. The IP address ranges you provide tell Discovery which specific devices to investigate. To retrieve useful information, Discovery needs credentials for devices within a particular range, so that Discovery can connect to and run various probes on the devices it finds. Discovery compares the devices it finds with configuration items (CI) in the CMDB and updates any matching devices. If Discovery does not find a matching CI in the CMDB, it creates a CI.
Discovery configuration procedures

Use the following links to configure Discovery for your environment. You do not need to perform all these procedures to run a Discovery. The platform provides many defaults you can use to explore your network that are suitable for most discoveries. To get started quickly with Discovery, you can use Guided Setup, which expedites the setup of a basic Discovery.

<table>
<thead>
<tr>
<th>Configuration procedures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery configuration</td>
</tr>
<tr>
<td>Discovery classifiers</td>
</tr>
<tr>
<td>Discovery status</td>
</tr>
<tr>
<td>Domain Separation and Discovery</td>
</tr>
<tr>
<td>Discovery classification parameters</td>
</tr>
<tr>
<td>Discovery Configuration Console</td>
</tr>
<tr>
<td>SNMP support for Discovery</td>
</tr>
<tr>
<td>Discovery identifiers</td>
</tr>
<tr>
<td>Discovery Dashboard</td>
</tr>
<tr>
<td>Discovery and SCCM together</td>
</tr>
<tr>
<td>Discovery properties</td>
</tr>
<tr>
<td>Discovery Home page</td>
</tr>
</tbody>
</table>

Discovery properties

Discovery properties allow you to control several aspects of the horizontal discovery process. Navigate to Discovery Definition > Properties to edit these properties:

**IP properties**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.enforce_ip_sync</td>
<td>Enforce syncing of IP addresses: This sets the first IP address. Each time a computer with multiple NICs is discovered, one of the IP addresses associated with the NICs is chosen as the IP Address field of the CI. A value of false collects all NIC IP addresses.</td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: true</td>
</tr>
<tr>
<td></td>
<td><strong>Default value</strong>: true</td>
</tr>
<tr>
<td>glide.discovery.exclude_ip_sync_classes</td>
<td>Defines CI classes whose IP addresses should not be substituted if the address returned by Discovery does not match one of the devices' NICs. Use a comma separated list to define multiple classes. By default, the system uses the management IP of a load balancer returned by Discovery in the CI record, rather than substituting it for the IP address of one of the load balancer's NICs.</td>
</tr>
<tr>
<td></td>
<td><strong>Type</strong>: string</td>
</tr>
<tr>
<td></td>
<td><strong>Default value</strong>: cmdb_ci_lb</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| glide.discovery.bgp_router_disable | BGP router exploration disable: Disables running the SNMP – Routing probe or pattern when discovering a router running the BGP protocol. Normally such exploration IS disabled because of the huge size of BGP routing tables, and because generally such routers are only operating at the edge of large networks where further network discovery would be irrelevant. This value could be set to false if your organization only uses BGP routers as edge routers between relatively small networks (such as between buildings on a single campus).  
  - **Type**: true | false  
  - **Default value**: true |
| glide.discovery.debug.network_discovery | Network discovery debugging: Enables extensive logging of all Network Discovery activities on the instance.  
  - **Type**: true | false  
  - **Default value**: true  
  - **Learn More**: Discovery monitoring and issue resolution |
| glide.discovery.discoverable.network.max.netmask.bits | Maximum netmask size for discoverable networks (bits): The maximum number of bits in a regular netmask for networks that are discovered by Network Discovery. A regular netmask is a netmask that can be expressed in binary as a string of ones followed by a string of zeroes. For example, 255.255.255.0 is regular and 255.255.255.64 is irregular. Regular networks are commonly expressed like: 10.0.0.0/24, which means a network address of 10.0.0.0 with a netmask of 255.255.255.0. Larger bit numbers mean networks with smaller numbers of addresses in them. For example, the network 10.128.0.128/30 has four addresses in it: one network address (10.128.0.128), one broadcast address (10.128.0.131), and two usable addresses (10.128.0.129 and 10.128.0.130). Small networks are commonly configured in network gear to provide loopback addresses or networks used strictly by point-to-point connections. Since these sorts of networks generally don’t need to be discovered by Network Discovery, it would be useful to filter them out. By setting this property to a value of 1 through 32, you can limit the sizes of regular networks that are discovered. Setting it to any other value causes all networks to be discovered. Irregular networks are always discovered. The default value is 28, which means that regular networks with 8 or fewer addresses cannot be discovered.  
  - **Type**: integer  
  - **Default value**: 28 |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.network_owner_method</td>
<td>Network router selection method: This property controls the method used to decide (during Network Discovery) which router should be selected as the router to be associated with a given IP Network. The possible values are:</td>
</tr>
<tr>
<td></td>
<td>• <strong>First Router</strong>: the first router that discovers the network is associated</td>
</tr>
<tr>
<td></td>
<td>• <strong>Last Router</strong>: the last router that discovers the network is associated</td>
</tr>
<tr>
<td></td>
<td>• <strong>Most Networks</strong>: the router with the most attached networks is associated</td>
</tr>
<tr>
<td></td>
<td>• <strong>Least Networks</strong>: the router with the least attached networks is associated</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: choice list</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: Most Networks</td>
</tr>
<tr>
<td></td>
<td>• <strong>Additional options</strong>: First Router, Last Router, Least Networks</td>
</tr>
<tr>
<td>glide.discovery.physical_interface_types</td>
<td>Physical interface types: A comma-separated list of interface types that are considered <em>physical</em> for the purposes of network discovery. If a router (or device capable of routing) has an interface of this type, the networks connected to that interface are considered locally connected to that device. The default interface types include Ethernet, 802.11, and Token Ring types. Interface type numbers are defined in the SNMP MIB-2, specifically in OID 1.3.6.1.2.1.2.2.1.3.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: string</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: 6,117,9,71,209</td>
</tr>
<tr>
<td>glide.discovery.switch_interface_types</td>
<td>Switch interface types: A comma-separated list of interface types that are considered. Interface type numbers are defined in the SNMP MIB-2, specifically in OID 1.3.6.1.2.1.2.2.1.3. Devices with any interface types that do not appear in this list are classified as routers, when they have routing. A complete list of the interface type numbers may be found on the IANA web site, in the section &quot;ifType definitions.&quot;</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: string</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: 7,8,9,26,53,62,69,71,78,115,117,209</td>
</tr>
</tbody>
</table>

### Other Discovery properties

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>com.snc.useExternalCredentials</td>
<td>Allows the use of external storage of credentials.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: true</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>glide.discovery.active_processes_filter</td>
<td>Active Processes Filter: Optimization for application dependency mapping. Filters the active processes returned by Discovery to only those that have a match in the Process Classification table.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: false</td>
</tr>
<tr>
<td>glide.discovery.adme.aggregation_interval</td>
<td>ADME - Aggregation interval (sec): How often to aggregate instantaneous data into chunks in seconds. It must be a minimum of 60 seconds and it must be a multiple of ADME Sampling interval.</td>
</tr>
<tr>
<td></td>
<td>* Type: integer</td>
</tr>
<tr>
<td></td>
<td>* Default value: 3600</td>
</tr>
<tr>
<td>glide.discovery.adme.base_dir_unix</td>
<td>ADME - Unix Base Dir: An existing directory on the target Unix machines to be used as a workspace. Must be an absolute path to the directory.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: false</td>
</tr>
<tr>
<td>glide.discovery.adme.base_dir_windows</td>
<td>ADME - Windows Base Dir: A network share on the target Windows machines to be used as a workspace.</td>
</tr>
<tr>
<td></td>
<td>* Type: string</td>
</tr>
<tr>
<td></td>
<td>* Default value: admin$\backslash\text{temp}$</td>
</tr>
<tr>
<td>glide.discovery.adme.max_total_samples</td>
<td>ADME - Max total samples: The limit of samples being taken by the script. The script will stop running after this amount of samples are taken.</td>
</tr>
<tr>
<td></td>
<td>* Type: integer</td>
</tr>
<tr>
<td></td>
<td>* Default value: 100</td>
</tr>
<tr>
<td>glide.discovery.adme.rolling_window_size</td>
<td>ADME - Rolling window size: How many recently aggregated chunks to keep. An aggregated chunk consists of ADME Aggregation interval divided by ADME Sampling interval samples. Only the newest chunks will be kept.</td>
</tr>
<tr>
<td></td>
<td>* Type: integer</td>
</tr>
<tr>
<td></td>
<td>* Default value: 24</td>
</tr>
<tr>
<td>glide.discovery.adme.sampling_interval</td>
<td>ADME - Sampling interval (sec): How often to sample process and connection data in seconds. Must be a minimum of 5 seconds.</td>
</tr>
<tr>
<td></td>
<td>* Type: integer</td>
</tr>
<tr>
<td></td>
<td>* Default value: 120</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>glide.discovery.application_mapping</td>
<td>Application mapping: Enable the application mapping portion of Discovery</td>
</tr>
<tr>
<td></td>
<td><strong>Attention:</strong> Disabling this property disables the creation of relationships between applications but does not disable the ADM probes and sensors, nor does it prevent process classification.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: true</td>
</tr>
<tr>
<td></td>
<td>* Learn More: Application Dependency Mapping (ADM) for Discovery</td>
</tr>
<tr>
<td>glide.discovery.debug.ci_identification</td>
<td>CI identification debugging: If set to true, enables debug logging (into the CI Identification Log) for CI Identification.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: false</td>
</tr>
<tr>
<td>glide.discovery.discover_aws_ec2_host_metadata</td>
<td>When doing IP-based discovery against a given host, also run probes that retrieve AWS EC2 metadata.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: true</td>
</tr>
<tr>
<td>glide.discovery.discover_azure_host_metadata</td>
<td>When doing IP-based discovery against a given host, also run probes that retrieve Azure metadata.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: true</td>
</tr>
<tr>
<td>glide.discovery.discover_software</td>
<td>Discover software packages: Enable the discovery of software packages.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: true</td>
</tr>
<tr>
<td></td>
<td>* Learn More: General software package discovery</td>
</tr>
<tr>
<td>glide.discovery.domain.name.nbt</td>
<td>Set OS domain name by NBT or WMI. If set to true, Windows domain name is set by NBT. Otherwise it is set by WMI.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: true</td>
</tr>
<tr>
<td>glide.discovery.enable_adme</td>
<td>ADME: Enable enhanced ADM probe. If set to true, the ADM Enhanced probe will be triggered and only fall back to the ADM probe as needed.</td>
</tr>
<tr>
<td></td>
<td>* Type: true</td>
</tr>
<tr>
<td></td>
<td>* Default value: false</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>glide.discovery.enable_file_tracking</td>
<td>Enable configuration file tracking as part of the Pattern based Horizontal Discovery. If set to <strong>true</strong>, the system allows you to specify paths of configuration files for applications that are discovered by patterns. You can track and compare tracked configuration files.</td>
</tr>
</tbody>
</table>
|                                           | - **Type**: true | false
|                                           | - **Default value**: true                                                                                                                  |
| glide.discovery.enable_mac_address_verification | Enables MAC address verification to determine if a device has changed its IP address during a discovery. If set to **true**, Discovery passes the MAC address of a device being discovered to the MID Server so that probes can determine if the IP address has changed while Discovery is running. If the device has changed IP addresses, then Discovery stops processing that IP address and updates the Discovery log with a warning message. Discovery does not stop processing other IP addresses. |
|                                           | - **Type**: true | false
|                                           | - **Default value**: false                                                                                                                  |
| Open the System Properties (sys_properties) table to find this property. |                                                                                                                                            |
| glide.discovery.enable.software_filter    | Windows software filter: Enable Windows software filtering for Discovery.                                                                       |
|                                           | - **Type**: true | false
|                                           | - **Default value**: true                                                                                                                  |
| glide.discovery.enforce_unique_ips        | Enforce unique IP addresses: Ignores the IP address after Discovery encounters subsequent devices that use the same IP address. Each time a computer, printer, or network gear with a valid IP address is discovered, any other devices with the same IP address have their IP address field cleared. If disabled, stores the IP address for each device. |
|                                           | - **Type**: true | false
<p>|                                           | - <strong>Default value</strong>: false                                                                                                                  |
| glide.discovery.file_tracking.max_file_size | File Tracking: The maximum file size for tracked configuration file content.                                                                     |
|                                           | - <strong>Type</strong>: integer                                                                                                                        |
|                                           | - <strong>Default value</strong>: 500000                                                                                                                  |
| glide.discovery.file_tracking.max_files_per_ci | File Tracking: The maximum number of tracked configuration files per CI.                                                                    |
|                                           | - <strong>Type</strong>: integer                                                                                                                        |
|                                           | - <strong>Default value</strong>: 50                                                                                                                     |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.discovery.file_tracking.change_limit_count | File Tracking: Number of changes allowed on a tracked configuration file in the defined time window.  
  - **Type**: integer  
  - **Default value**: 4 |
| glide.discovery.file_tracking.change_limit_days | File Tracking: Time window (in days) that limits the number of changes on a tracked configuration file.  
  - **Type**: integer  
  - **Default value**: 7 |
| glide.discovery.fqdn.regex | DNS Host Name And Domain Name Regex: The default parsing of FQDN (Fully Qualified Domain Name) is to pick the first name separated by dots as the host name and the rest of the names as the domain name. For example, “machine1.testlab.service-now.com” has host name of “machine1” and domain name of “testlab.service-now.com”. The property allows regex with two capturing groups with the first group representing the host name and the second group the domain name.  
  - **Type**: string  
  - **Default value**: `^(([^.]+)\.(?:[^.]+)+[^.]+$` |
| glide.discovery.hostname.always_update | Always update host name. If set to **true**, discovery always updates the host name with the most recently discovered value contingent upon the source being trusted. Note that this may result in hand-entered values being overwritten.  
  - **Type**: true | false  
  - **Default value**: true |
| glide.discovery.hostname.case | Host name case. If **Lower case** is selected, always translate the host name into lower case; if **Upper case** is selected, always translate the host name to upper case; if **No change** is selected, leave the host name intact. This primarily affects host names discovered with NETBIOS, though some non-standard DNS systems may also return some or all of the name in upper case.  
  - **Type**: choice list  
  - **Default value**: Lower case  
  - **Additional options**: Upper case, No change |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.discovery.hostname.dns_nbt_trusted | DNS or NetBIOS is trusted host name source: If set to **true**, trust the device name discovered via DNS or NetBIOS. If checked, CI's host name found via DNS or NBT is used.  
  - **Type**: true | false  
  - **Default value**: true |
| glide.discovery.hostname.include_domain | Includes domain name in host name: If set to **true**, include the domain name as part of the host name. For example, "bosco.service-now.com" instead of "bosco".  
  - **Type**: true | false  
  - **Default value**: false |
| glide.discovery.hostname.snmp_trusted | SNMP is trusted host name source: If set to **true**, trust the device name discovered via SNMP. If checked, any device name found via SNMP is used instead of the name found by a reverse DNS lookup.  
  - **Type**: true | false  
  - **Default value**: false |
| glide.discovery.hostname.ssh_trusted | SSH is trusted host name source: If set to **true**, trust the device name discovered via SSH. If checked, any device name found via SSH is used instead of the name found by a reverse DNS lookup.  
  - **Type**: true | false  
  - **Default value**: false |
| glide.discovery.hostname.wmi_trusted | WMI is trusted host name source: If set to **true**, trust the device name discovered via WMI. If checked, any device name found via WMI is used instead of the name found by a reverse DNS lookup.  
  - **Type**: true | false  
  - **Default value**: false |
| glide.discovery.ip_service_affinity | IP service affinity: If set to **true**, IP service affinity is enabled. IP service affinity allows Discovery to remember the last port of the IP address that was discovered.  
  - **Type**: true | false  
  - **Default value**: false  
  - **Learn More**: [Check IP service affinity for Discovery and Orchestration](#)
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.discovery.L3_mapping | Map servers and network devices to routers and layer-3 switches if the "L3 mapping" property is enabled, it will map servers and network gears to its associated routers and layer-3 switches.  
- **Type**: true | false  
- **Default value**: true  
- **Learn More**: [Network discovery](#) |
| glide.discovery.log_message_length | Log Message Length: Limit the maximum message length that is displayed in Discovery Log table. A value of 0 or any negative number disables this limit.  
- **Type**: integer  
- **Default value**: 200  
- **Learn More**: [Logs for Discovery](#) |
| glide.discovery.max_concurrent_invocations_per_schedule | Maximum concurrent invocations per schedule: Prevents an unbounded number of invocations from inundating the system when a schedule takes longer than the time between invocations. The value is an integer defining the maximum number of automated invocations of the same schedule that may proceed at one time. If the limit has been reached subsequent scheduled invocations are cancelled. The default value is 3. A value of 0 or any negative number disables this restriction.  
- **Type**: integer  
- **Default value**: 3 |
| glide.discovery.network_discovery.functionality | Networks discovery functionality: the Functionality used to discover networks. Usually this should be **SNMP only**.  
- **Type**: string  
- **Default value**: SNMP only |
| glide.discovery.perf.metrics.enable_collection | Whether or not to collect Discovery performance metrics.  
- **Type**: true | false  
- **Default value**: true |
| glide.discovery.perf.metrics.rollup_by_build | Whether or not to aggregate individual Discovery performance metrics into a roll-up table that groups by discovery build.  
- **Type**: true | false  
- **Default value**: true |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.discovery.perf.metrics.rollup_by_status | Whether or not to aggregate individual Discovery performance metrics into a roll-up table that groups by discovery status.  
  - **Type**: true | false  
  - **Default value**: true |
| glide.discovery.perf.metrics.rollup_by_target | Whether or not to aggregate individual Discovery performance metrics into a roll-up table that groups by target IP address.  
  - **Type**: true | false  
  - **Default value**: true |
| glide.discovery.populate_software_instance_table | If set to true, Discovery populates these tables for software discovery:  
  - cmdb_software_instance  
  - cmdb_sam_sw_install  
  - **Type**: true | false  
  - **Default value**: false  
  
  Use this property only if you were using Software Asset Management with Discovery in pre-Jakarta releases and want to keep populating the cmdb_software_instance table. Open the System Properties (sys_properties) table to find this property. |
| glide.discovery.roundingInterval.cpu | CPU speed rounding: Enter the number to round the CPU speed to. The units are in MHz.  
  - **Type**: integer  
  - **Default value**: 1 |
| glide.discovery.roundingInterval.ram | Memory rounding: Enter the number to round the computer RAM to. The units are in MB.  
  - **Type**: integer  
  - **Default value**: 1 |
| glide.discovery.sensors.fire_ssh_probe | In case the VIPs (Load Balancer Virtual Services) did not return as part of the SNMP Probe payload, fire the SSH Probe. Select this option to use the SSH port on the load balancer, rather than just using SNMP.  
  - **Type**: true | false  
  - **Default value**: true |
<p>| glide.discovery.sensors.netapp_native_7mode | For 7-Mode NetApp storage servers use native discovery instead of SMI-S. |
| glide.discovery.sensors.netapp_native_cluster_mode | For Cluster Mode NetApp storage servers use native discovery instead of SMI-S. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.sensors.save_attachments</td>
<td>Save ECC queue attachments: The normal behavior for discovery sensors is to delete attachments to ECC queue entries upon successful sensor processing. Setting this property to true overrides this behavior, and forces attachments to be preserved. This would normally only be useful for debugging purposes.</td>
</tr>
</tbody>
</table>
|                                           | • **Type**: true | false  
|                                           | • **Default value**: true                                                                                                                                                                                      |
| glide.discovery.shazzam_ranges_json       | Use JSON for IP ranges in Shazzam: If set to **true**, discovery will encode Shazzam’s IP ranges as JSON, dramatically reducing the payload size.                                                                 |
|                                           | • **Type**: true | false  
|                                           | • **Default value**: true                                                                                                                                                                                      |
| glide.discovery.software_filter_keys      | Windows software filters: If Windows software filtering is enabled, when the name of a discovered software contains any of the comma separated values, it will be filtered out.                                           |
|                                           | • **Type**: string |
|                                           | • **Default value**: Language Pack, Hotfix Security Update                                                                                                                                                      |
| glide.discovery.software_sccm_managed     | Windows software is SCCM managed: If set to **true**, Discovery will not populate software for computer CIs also managed by SCCM.                                                                                   |
|                                           | • **Type**: true | false  
|                                           | • **Default value**: false                                                                                                                                                                                      |
|                                           | • **Learn More**: [Discovery and SCCM together](#)                                                                                                                                                           |
| glide.discovery.use_cmdb_identifiers      | CMDB Identifiers: If set to **true**, identification and reconciliation will be handled by the CMDB API instead of through the old Discovery implementation.                                                   |
|                                           | • **Type**: true | false  
|                                           | • **Default value**: true                                                                                                                                                                                      |
| glide.discovery.use_probe_results_cache   | Use probe results cache: If set to **true**, the cache is checked to see if the results of the probe need to be processed by a sensor. It only needs to be processed if the results have changed from the last discovery run. |
|                                           | • **Type**: true | false  
|                                           | • **Default value**: false                                                                                                                                                                                      |
|                                           | • **Learn More**: [Create or modify a probe](#)                                                                                                                                                              |

*Note:* In previous releases, this property was **true** by default. The value in this property is not changed during an upgrade to Kingston.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.discovery.virtual_interface_types | Virtual interface types: List of interface types (comma-separated) that can be considered "virtual" for the purposes of network discovery. In other words, if a router (or device capable of routing) has an interface of this type, the networks connected to that interface are considered virtually connected to that device. The default interface types include the propVirtual type. Interface type numbers are defined in the SNMP MIB-2, specifically in OID 1.3.6.1.2.1.2.2.1.3.  
  - **Type**: integer  
  - **Default value**: 53 |
| glide.discovery.warn_minor_version | Warn on Minor Version Mismatch: If set to true, warnings are logged when minor_version mismatches are detected during Discovery sensor processing.  
  - **Type**: true | false  
  - **Default value**: false |
| glide.ecc_agent.validated.override | An override for the ecc_agent validated field: If set to true, ecc_agent records will always be treated as valid.  
  - **Type**: true | false  
  - **Default value**: false  
Navigate to System Properties > All Properties to find this property. |
| glide.eccprobe.awsrestprobe.max_retries | Set the maximum number of times the AWS request will be retried if a rate limit exceeded error occurs. Maximum value is 10. Note that retries use exponential backoff (2^x). Setting negative values is equivalent to setting the maximum value.  
  - **Type**: integer  
  - **Default value**: 4 |
| glide.eccprobe.awsrestprobe.max_wait | Set the maximum total amount of time (seconds) the AWS request attempts to retry. Maximum value: 2046 seconds. Setting negative values is equivalent to setting the maximum value.  
  - **Type**: integer  
  - **Default value**: 30 |
| glide.eccprobe.max_queued_probes_per_run | Set the maximum number of probes a node can pick up during a run.  
  - **Type**: integer  
  - **Default value**: 20 |
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.eccprobe.node_agent_id</td>
<td>ECC queue entries with ‘mid.server.$(VALUE)’ will be picked up and processed by the nodes. This must be unique and not the same as any MID server name.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: string</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: NODE_AGENT</td>
</tr>
</tbody>
</table>

### Discovery IP address configuration

Use one or more of these methods in any combination to define the network or network segment for Discovery to query. You can include or exclude specific IP ranges from your query.

**Note**: If you do not know the IP addresses in the network, run Network discovery first to determine the IP networks. Then, convert the IP networks into IP address range sets.

There are three types of IP collections:

<table>
<thead>
<tr>
<th>IP collection type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address list</td>
<td>Use IP address lists to add individual addresses to query. These addresses are not included in any existing IP range or IP network. You can enter the IP address of the device or a host name (DNS name). If you enter a host name, it must be mapped to an IP address.</td>
</tr>
<tr>
<td>IP address range</td>
<td>You can define arbitrary ranges of IP addresses to query. This process is a good way to include selected segments of a network or subnet. However, Discovery has no way of knowing if the IP range includes addresses for private networks or broadcast addresses, and scans all the addresses in the range. If the network and broadcast addresses are included, then the results are inaccurate. Discoveries configured to detect IP networks are more accurate than discoveries configured for IP address ranges. Only those IP addresses in your range that are reserved for manageable devices on the public network should be included.</td>
</tr>
</tbody>
</table>

**Note**: To avoid any possible performance issues, limit Discovery schedules to a maximum range of /16 or no more than 65,000 IPs. You can also split the Discovery schedules into even smaller ranges of IP addresses.
### IP collection type

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP network</td>
</tr>
</tbody>
</table>

You can also scan an entire IP network. An IP network includes the range of available IP addresses in that network. The scan also includes the network address (the lowest address in the range) and the broadcast address (the highest address in the range). After you run network discovery, convert the IP networks that were found into range sets for use in discovering other devices.

An example of a class C network range is 192.168.0.0 to 192.168.0.255. In the Range Set form, this network can be entered with either of the following notations:

- 192.168.0.0/24
- 192.168.0.1/255.255.255.0

This notation indicates that Discovery is scanning an IP network, and does not scan the highest and lowest numbers in the range. Significant errors are prevented and introduced into the Discovery data by the broadcast address. All devices are returned in the network, and the network address, which can add an arbitrary number of redundant devices. This built-in control makes IP networks the best method of defining which IP address ranges to query.

### IP address selection properties

You can use system properties to control the selection of IP address for specified CI classes.

Use these properties to determine if the system should replace the IP address returned by Discovery in a device’s CI record if the address does not match that of a network interface (NIC) on the device. This is important for the discovery of devices with management IP addresses that differ from IP addresses associated with one or more NICs on the device. Because a device’s management IP is used in the Discovery schedule for that device, this is the address that Discovery returns. Use these properties to determine which IP address to use for CIs of any class.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.enforce_ip_sync</td>
<td>Enforce syncing of IP addresses: This sets the first IP address. Each time a computer with multiple NICs is discovered, one of the IP addresses associated with the NICs is chosen as the IP Address field of the CI. A value of false collects all NIC IP addresses.</td>
</tr>
<tr>
<td></td>
<td>Type : true</td>
</tr>
<tr>
<td></td>
<td>Default value: true</td>
</tr>
</tbody>
</table>
### Field: glide.discovery.exclude_ip_sync_classes

**Description:** Defines CI classes whose IP addresses should not be substituted if the address returned by Discovery does not match one of the devices’ NICs. Use a comma separated list to define multiple classes. By default, the system uses the management IP of a load balancer returned by Discovery in the CI record, rather than substituting it for the IP address of one of the load balancer’s NICs.

- **Type:** string
- **Default value:** cmdb_ci_lb

---

**Create a Quick IP range for a Discovery schedule**

Quick ranges allow administrators to define IP addresses to scan in a single comma-delimited string without creating separate records.

Only MID Servers that are up and validated are used with quick ranges. The MID Servers must specify the Discovery application (or ALL applications) and have IP ranges configured if you use the auto-select feature on the Discovery schedule.

You can enter IP addresses in one of the following formats:

- An IP range defined by a slash and the number of bits in the subnetwork. For example, the string 10.10.10.0/24 scans 24 bits of IP addresses from 10.10.10.0 to 10.10.10.254.
- An IP range defined by a dash. For example, the string 10.10.11.0-10.10.11.165 scans the IP addresses from 10.10.11.0 to 10.10.11.165.
- A comma-separated list of specific IP addresses. For example, the string 10.10.11.200,10.10.11.235 scans the IP addresses 10.10.11.200 and 10.10.11.235.

1. Click the **Quick Ranges** related link on the Discovery Schedule form.
2. Enter the IP ranges and specific IP addresses to scan.
3. Click **Make Ranges**.

**Note:** The Quick Range interface is for entering IP addresses only and cannot be used to edit IP addresses that have already been submitted.
Entering a quick range from a Discovery schedule

Advanced

- Include alive
- Log state changes
- Use SNMP version v1/v2

Quick Ranges

Enter comma-separated IP address ranges, IP networks, or individual IPs describing the range to add. For example:

```
10.0.1.0/24,10.0.2.1-10.0.2.15,10.0.3.176,10.0.3.222
```

Specifies an IP network with valid IP addresses between 10.0.1.1 and 10.0.1.254 inclusive, and an IP range from 10.0.2.1 to 10.0.2.15 inclusive, and the two individual IP addresses 10.0.3.176 and 10.0.3.222 you make that cannot be interpreted will simply be ignored.

```
10.0.1.0/24,10.0.2.1-10.0.2.15,10.0.3.176,10.0.3.222
```

Related Links

Quick ranges
Discover now
The instance automatically displays the entries in the proper format.

4. For changes to IP address ranges, select the IP address records in the **Discovery Range Sets** related list.

**Import IP ranges into Discovery schedules with import sets**

The most efficient method of entering large numbers of IP networks into Discovery schedules is by using import sets.

Common groups of IP addresses, known as **ranges** can be used in Advanced Discovery schedules.

Use a data source that can be mapped. Include these fields:

- Start IP: the first IP address in the range (inclusive).
- End IP: the last IP address in the range (inclusive).

1. Navigate to **System Import Sets > Load Data**.
2. Identify the file or data source that contains the desired information.
3. Create a table name, such as `ipnetworks`.
4. Select **Upload an Excel file** and browse to the source file.
5. Click **Go** to import the file.

![Image of System Import Sets](image)

**Data Source**

6. Navigate to **System Import Sets > Create Transform Map** and map the items in the Excel spreadsheet to the fields of the CMDB in the target table IP Range `[ip_address_range]` table.
7. Give the Transform Map a unique and descriptive name.
8. Submit the form, and then click **New** in the **Field Maps** Related List.
9. Map the fields from the Excel spreadsheet to the fields in the IP Range `[ip_address_range]` table. The fields you need values for are the Start IP and End IP addresses.
10. Click the **Mapping Assist** Related Link and use the lists that appear to resolve the fields between the table and the data source (the Excel spreadsheet in this example).
11. Click **Save**.

The view returns to the Table Transform Map form.
12. Click **Transform** in the Related Links to move the data into the proper fields in the IP Range `[ip_address_range]` table.
The imported IP ranges are available now for use in any advanced Discovery schedule.

**Exclude IP ranges from a Discovery range set**
You can specify a range of IP addresses that you want to exclude from your Discovery query.

1. Navigate to **Discovery > Discovery Range Sets**.
2. Select one of the range sets.
3. Select one of the types from the **Discovery IP ranges** related list.
4. Click **New** from the **Discovery Range Item Excludes** related list.
5. Enter a starting IP and ending IP address to create the range to exclude.
6. Click **Submit**.

**Domain Separation and Discovery**
This is an overview of domain separation and Discovery. Domain separation enables you to separate data, processes, and administrative tasks into logical groupings called domains. You can then control several aspects of this separation, including which users can see and access data.

**Overview**

**Support: Level 2**
ServiceNow applications are defined by incremental support levels. The Discovery application includes all of the features of lower support levels.

Domain separation is supported in this application. Not all ServiceNow applications support domain separation; some include limitations on the data and administrative settings that can be domain separated. To learn more, see Application support for domain separation. Application support for domain separation.

Service Providers (SPs) use domain separation to segregate data for each customer. Users in a given domain have visibility only to the data in their own domains or in child domains. SPs typically control the top-level domain, which gives them visibility to data associated with all domains. Given that Discovery domain separation support is considered Level 2, there is no delegated administration to the child domains. The SPs must retain administrative control.

**How domain separation works in Discovery**

Multiple domains can be supported by a single MID Server. In releases prior to Kingston, each MID Server could support only a single domain. In newer releases, segregating domains by MID Server is useful when the domain is large, or when the domain's resources are held in a customer's data center rather than the SP's. For Discovery on MID Servers supporting a single domain, the discovered CIs are assigned to the domain of the MID user used to authenticated against the ServiceNow instance. In multi-domain MID Servers, the discovered CIs are assigned to the domain of the user that created the Discovery schedule.

Discovery implements data domain separation through the MID Server by impersonating the MID Server user during sensor processing. Discovery uses the domain that the MID Server user is in to determine which domain the discovered data should be put into. Discovery configuration information, including classifiers, identifiers, probes, and sensors, is not domain-separated.

SPs generally use IP-based Discovery. In cases where the SP controls the network addressing, they divide the address space among their customers to ensure that each domain has a distinct IP address space. The SP assigns one or more subnets to a customer or domain and creates Discovery schedules for those subnets.
If the SP is remotely managing their customer's data center, there will often be some overlap between address spaces different customers use. In these cases, the SP can use network address translation (NAT) on the IP range and run a Discovery schedule.

Once the CIs are assigned to the correct domain, the visibility and read/write access control are provided by the platform through the domain hierarchy. Schedules are visible to users in their respective domains. Cross-domain schedule visibility is not possible, except for the SP who controls the parent domain and has visibility to all domains.

**Domain separation for MID Server files**

You can create versions of these specific MID Server policy records that only a MID Server from the same domain can use. This process separation is supported for records in tables that extend MID Server Synchronized Files (`ecc_agent_sync_file`):

- MID Server MIB File (`ecc_agent_mib`)
- MID Server JAR File (`ecc_agent_jar`)
- MID Server Script File (`ecc_agent_script_files`)

By default, all records in these tables are members of the global domain. A user can override the default global domain and create a version of these policies for use in the user's own domain.

**Note:** Attachments on MIB or JAR file records might not appear as they did in a non-domain separated environment. The attachments do not appear because the Attachments (sys_attachment) table is data separated. When data is separated between domains, a record in a child domain cannot access records in a parent domain.

See [MID Server domain separation](#) for instructions on setting up domain separation through the MID Server.

**Domain-separated tables**

Records in all tables that extend the Base Configuration Item (cmdb) table can be domain separated. In addition, records in these tables can also be domain separated:

- Serial Number (`cmdb_serial_number`)
- TCP Connection (`cmdb_tcp`)
- Fibre Channel Initiator (`cmdb_fc_initiator`)
- Fibre Channel Targets (`cmdb_fc_target`)
- IP Address to DNS Name (`cmdb_ip_address_dns_name`)
- Service (`cmdb_ip_service_ci`)
- KVM Virtual Device (`cmdb_kvm_device`)
- Load Balancer Service VLAN (`cmdb_lb_service_vlan`)
- Load Balancer VLAN Interface (`cmdb_lb_vlan_interface`)
- Switch Port (`cmdb_switch_port`)

**SNMP support for Discovery**

Discovery supports SNMP versions 1, 2c, and 3. Discovery uses version 1 and 2c by default. You must enable support for version 3.

MID Servers support all SNMP protocol versions by default. You can set a MID Server to support only specific versions of SNMP.
Add an SNMPv3 user credential in Discovery
Set up your credentials so Discovery can access SNMPv3 targets.

Role required: admin

A MID Server parameter is also available to control SNMP versions. See MID Server parameters.

1. Navigate to Discovery > Credentials.
2. Click New.
3. Click SNMPv3 Credentials.
4. Use the following table to fill in the form.
   This form creates an entry in the SNMPv3 credentials table. The SNMPv3 credentials table stores the credentials.

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique and descriptive name for this credential.</td>
</tr>
<tr>
<td>Field</td>
<td>Input Value</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Active</td>
<td>If the credential can be used. Select this box to activate the credential.</td>
</tr>
<tr>
<td>User name</td>
<td>The user name to authenticate to the target devices. This selection must match the name that is set on the devices to be discovered. If a domain account is used to execute PowerShell commands in Discovery, the user name must include the domain. Avoid leading or trailing spaces in user names. A warning appears if the platform detects leading or trailing spaces in the user name. This is the only required field.</td>
</tr>
</tbody>
</table>
| Authentication protocol | The encryption protocol. This selection must match the encryption protocol that is set on the devices to be discovered. Valid options are:  
  - **None**: No encryption is used.  
  - **MD5** *(Message Digest version 5)*  
  - **SHA** *(Secure Hash Algorithm)*  
  See the SNMPv3 security options table below to configure the type of security.                                                                                           |
| Authentication key    | The authentication key that is set on the target devices.                                                                                                                                                     |
| Privacy protocol     | The encryption standard. This selection must match the encryption standard that is set on the devices to be discovered. Valid options are None, AES128 *(standard AES)*, AES192, AES256, DES, or 3DES.                                                |
| Privacy Key          | The privacy key that is set on the target devices.                                                                                                                                                           |
| Tag                  | The name of a credential tag.                                                                                                                                                                                   |
| Applies to           | The MID servers that use this credential. You can choose **All MID servers**, or you can create a list of **Specific MID servers**.                                                                           |
| Order                | The order in which the credentials are tried, if there is more than one credential for a given type.                                                                                                             |

**SNMPv3 security options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Fields to configure</th>
</tr>
</thead>
<tbody>
<tr>
<td>noAuthNoPriv security</td>
<td></td>
</tr>
</tbody>
</table>
  - **Authentication protocol**: None  
  - **Privacy protocol**: None  
  - **Authentication Key**: Leave blank  
  - **Privacy Key**: Leave blank                                                                 |
Option | Fields to configure
--- | ---
authNoPriv security | - Authentication protocol: Set
                  - Privacy protocol: Leave blank
                  - Authentication Key: Set
                  - Privacy Key: Leave blank

authPriv | - Authentication protocol: Set
          - Privacy protocol: Set
          - Authentication Key: Set
          - Privacy Key: Set

Set SNMP version on the Discovery schedule
You must also add an SNMPv3 credential if you set the SNMP version to v3 or All.

Role required: admin
A MID Server parameter is also available to control SNMP versions. See MID Server parameters.

1. Navigate to Discovery > Discovery schedules > your Discovery schedule.
2. In the Use SNMP Version field, select the appropriate version.
   Valid options are v1/v2c, v3, or All.

Discovery and SCCM together
Use these guidelines to avoid common issues when you use Discovery and System Center Configuration Manager (SCCM) together.

You must decide if you will use Asset Intelligence (AI) or non-AI software data sources. Because the software correlation data is tracked differently for each of these data sources, you should avoid switching back and forth between the two. If you must switch between data sources, remove all the software installation records imported by SCCM before the switch occurs.

When Software Asset Management is not enabled, software installation records are stored in the Software Instance (cmdb_software_instance) table. When Software Asset Management is enabled, software installation records are stored in the Software Installation (cmdb_sarn_sw_install) table.

Note: If records are not removed before the switch, duplicate records may exist. In the event that AI and non-AI data becomes mixed, clear the Software Installation table.

Overview of SCCM plugins
These are the plugins for the currently supported SCCM versions, all of which support Asset Intelligence and Incremental Software Reconciliation:
- Microsoft SCCM 2007
- Microsoft SCCM 2012 v2
- Microsoft SCCM 2016
Upgrading to a new version

The ServiceNow SCCM integrations are self-contained and can exist independently. They each use their own import set tables, data sources and transform maps. However, all SCCM integrations will transform data into the same tables within the ServiceNow CMDB. To avoid the data being overwritten by another source:

- Use one SCCM integration and disable all other SCCM scheduled imports.
- Perform a full import to clear the cmdb_software_instance table, the cmdb_sam_sw_install table, and other tables of old SCCM data.

To change the SCCM integration:

- Disable the current integration by deactivating the SCCM import schedule.
- Activate the new SCCM plugin.
- Reimport all the software records when you are switching to an integration that supports incremental imports of removed software.

For instructions on disabling the current SCCM import schedule, see Upgrade the SCCM integration version

Collect software data with either SCCM or Discovery

When Discovery and SCCM are both enabled on a system, the software records found through both tools could overwrite each other.

Role required: admin

Determine how software data is collected.

1. Navigate to Discovery Definition > Properties.
2. Set the Windows software is SCCM managed property (glide.discovery.software_sccm_managed) appropriately.
   - If this property is set to Yes, software for computer CIs is populated in the CMDB by SCCM. An internal process determines that the computer is managed by SCCM, and Discovery does not populate the software records for Windows software data. The following information level message is written to the Discovery log file.
     Skipping software population because the CI is managed by SCCM
   - If the check box is cleared, the property is set to No, and the system manages software data with Windows.
3. Click Save.

Important: If SCCM Integration is active before Discovery is enabled and the property is enabled, Discovery ignores the population of software for any CIs that are also
imported through SCCM. If Discovery is enabled before the SCCM Integration, it is possible for software installation data from both sources to be mixed.

Discovery classifiers

A classifier tells Discovery which probes to trigger for the identification and exploration phases of discovery. Classifiers can also trigger the Horizontal Pattern probe, which launches a pattern, rather than additional probes, for identification and exploration.

The classifier essentially starts the identification stage. Discovery uses it after the classification probe returns important parameters to the instance that tell Discovery what to do next. In most cases, you do not need to create a classifier or modify a classifier. But if you are having trouble with horizontal discovery, you might want to check the conditions that determine when a classifier runs based on the parameters the classification probe returns to the instance. Or if you want to discover a new type of CI that Discovery does not already find, you can create your own classifier.

Device, process, and IP address classification

Discovery classification can be broken down into three types: device classification, process classification, and IP address (or IP scan) classification:

Device classification

The classification of actual device types, such as a computer running Windows, a computer running a flavor of UNIX or LINUX, a router, a switch, or a load balancer, and so on.

When Discovery identifies a computer CI, it triggers an active processes probe to explore the computer CI further. Discovery compares the results of the active processes probe to the process classification conditions to determine if there is a match.
Starting with the Madrid release, the horizontal discovery process can classify devices using HTTP. Of all protocols that Discovery uses (including WMI, SSH, and SNMP), HTTP is the lowest priority by default. Discovery uses HTTP classification only if:

- Shazzam determines that the ports for HTTP (80) and HTTPS (443) traffic are open.
- The horizontal discovery process fails for the higher priority port probes (WMI, SSH, and SNMP) if the ports for those protocols are not open, or if discovery for those protocols fail. The horizontal
discovery process can fail, for example, if SSH and SNMP credentials are not configured or are incorrect.

**Note:** See [Port Probes](#) for more details on how port scanning works and to see the priorities of the different protocols.

HTTP classification launches the [HTTP Classify](#) probe to classify the device. The [HTTP - Classify](#) probe runs a GET request for each device for each HTTP classification. The URL of the request is built as follows: `PROTOCOL://IP:PORT/PATH`, where:

- `PROTOCOL` is port 80 for HTTP or port 443 for HTTPS.
- `IP:PORT` is the IP address and port number of the device.
- `PATH` is the path that you specify in the HTTP classifier.

For more information on port probes, see [Port probes](#). For instructions on creating an HTTP classifier, see [Create an HTTP classification](#).

### Process classification

The classification of applications based on the processes that are running.

Discovery classifies processes during the last phase of discovery: the exploration phase, after identifying devices in the Computer (cmdb_ci_computer) table and its extensions. Just like device classification, process classification has its own classification criteria and also has the ability to launch probes. Unlike device classification, process classification creates child configuration items (CI) with Runs on::Runs relationships. By default, Discovery includes classifications for most common processes.

If a process matches the classification criteria, Discovery determines whether to run the process handler script. The process handler script modifies the parameter data to help Discovery identify whether the process represents an existing or new application CI. Discovery process handlers prevent the creation of duplicate CIs by filtering out parameters known to have inconsistent values before process classification occurs. Every time Discovery adds or updates an application CI, it also determines the application dependency mapping of the application CI to other CIs in the CMDB.
Process classification workflow

**IP address (IP scan) classification**

IP address discovery is credential-less, meaning that it attempts to identify devices and software based on just the open ports and banners it finds without requiring you to create credentials.
If the classification criteria are met for a device in the IP Scan mode, Discovery automatically updates the CI in the CMDB. After a device is properly classified, Discovery launches the exploration probes configured for that class of device and begins gathering detailed information about the CI.

In the default Discovery system, the Linux classifier triggers eleven exploration probes that return information such as disk size, memory, and the number of current connections. The data from these probes returns at different times and is stored in the ECC Queue until processing is complete.

This diagram shows the processing flow for classifying and probing devices with an IP scan (no identifiers):
See Classification for IP address discovery for more details about the parameters available to classifiers for this type of discovery.

**Classifier criteria**

Classifiers also provide criteria that you can use to specify when Discovery should use the classifier under the conditions that you define. The criteria is based on the parameters that a classify probe returns to Discovery. Criteria is constructed with the parameter, an operator, and a value.

**Note:** Condition filters in process classifiers are case-sensitive.

**Classifiers and patterns**

Discovery can use patterns, rather than probes, to identify and explore CIs. Discovery triggers patterns from the Horizontal Discovery probe, which can be specified on a classifier. You can create your own patterns and add them, via the Horizontal Discovery probe, to a classifier. See Add the Horizontal Pattern probe to a classifier for instructions. You might already be using one of the out-of-box patterns that are provided with Discovery. You can verify this by looking at the classifier to see if the Horizontal Pattern Probe is specified.

**Logging classification debugging information**

To log debugging information about classifications, add the following system property. The resulting log entries list the name of each classifier that runs, along with all the names and values that are available to the criteria in the classifier.

<table>
<thead>
<tr>
<th>System property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.debug.classification</td>
<td>Enables debugging information for process classification.</td>
</tr>
<tr>
<td></td>
<td>Type: true</td>
</tr>
<tr>
<td></td>
<td>Default Value: false</td>
</tr>
<tr>
<td></td>
<td>Location: Add to the System Properties (sys_properties) table</td>
</tr>
</tbody>
</table>

What you can do with Discovery classification

- Create or modify a discovery classifier if you want to classify CIs that Discovery does not already classify, or trigger other probes that are not already on a classifier. You can modify classifiers that Discovery uses in standard CI discovery, process classifiers for applications, and classifiers based on IP address scans.

Before you modify any classifiers, review the parameters that are available for each type of classifier.

- If Windows machines are on your network, you can use the WinRM protocol, rather than WMI, for more efficient lightweight data transfer and remote command execution. By default, Discovery uses WMI. For instructions on the classifier modifications you can make to use WinRM, see Use Windows Remote Management for classification.

- If you have Windows computers that are acting as servers and you want them to be classified by their function rather than by the operating system, you can make changes to the criteria
of the Windows classifier. See *Reclassify a Windows Workstation machine as a server* for instructions.

Create a Discovery CI classification

A CI classification allows Discovery to discover most common operating systems, network devices, and processes.

Role required: admin

**Caution:** Adding SNMP OID classifications with a Net-SNMP module OID starting with 1.3.6.1.4.1.8072.3.2 can result in misclassification of devices and incorrect model names. These are template system OIDs that can remain on a device if the manufacturer fails to replace them with OIDs specific to their own model. For example, a Check Point Smart-1 3050 Firewall, which uses Linux as its OS, might have the OID for a Linux server, 1.3.6.1.4.1.8072.3.2.10. This results in the firewall being classified incorrectly. If you know you only have one device with a specific template system OID, it is safe to add it. Discovery will always choose the classification matched to that OID. If two or more devices have the same OID, configure classification criteria that differentiate the devices from one another. For details, see KB0751287.

1. Navigate to **Discovery Definition > CI Classification**.
2. Select the type of classification to create.
3. Click **New** in the list of classifications for the type you selected.
4. Complete the form using the fields in the table.

**Classification form fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the configuration item (CI).</td>
</tr>
<tr>
<td>OID</td>
<td>SNMP system OID for matching this device. When the OID value matches that of an SNMP device, Discovery uses the information to populate the CMDB with the specified manufacturer, model, and classifier.</td>
</tr>
<tr>
<td>Operator</td>
<td>Operator for determining how to match and SNMP OID. The choices are <em>Is</em> and <em>Starts with</em>. This field is available only for SNMP System OIDs classifiers.</td>
</tr>
<tr>
<td>Active</td>
<td>Enables or disables this classifier. When a classifier is disabled, the system stops classification at this level and does not launch classifiers of a lower order. For example, when the classifier for Windows 2008 Server is disabled, the system stops Discovery at this point and does not launch the Windows 2012 Server classifier.</td>
</tr>
<tr>
<td>Order</td>
<td>Order (sequence) in which the platform runs this classifier.</td>
</tr>
<tr>
<td>Table</td>
<td>Table for this classification. For example, if this record classifies a Linux server, select the Linux Server (cmdb_ci_linux_server) table.</td>
</tr>
<tr>
<td>Classifier</td>
<td>Classification of an SNMP device, such as <em>A10 Load Balancer</em>. This field is available only for SNMP System OIDs classifiers.</td>
</tr>
<tr>
<td>Field</td>
<td>Input Value</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Relationship type</td>
<td>Type of relationship for this classifier, such as <strong>Runs on::Runs</strong>. This field is available only for application and process classifiers.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Criteria that must match to classify this device. The choices are <strong>Any</strong> of the parameters or <strong>All</strong> of the parameters.</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Name of the manufacturer of a network device. This field is available only for SNMP and SNMP System OIDs classifiers.</td>
</tr>
<tr>
<td>Model</td>
<td>Model number of a network device. This field is available only for SNMP and SNMP System OIDs classifiers.</td>
</tr>
</tbody>
</table>
| On classification script    | Script that runs if classification criteria are met. Use this script to perform any special tasks after a device is classified. It is possible to use the `g_probe_parameters` hashmap from within a classification script to set probe parameters for any configured, triggered probes. For example, this code sets a `node_port` parameter to 16001 for all triggered probes:  

```javascript
(JS),
    g_probe_parameters['node_port'] = 16001; //
```

<table>
<thead>
<tr>
<th>Related lists</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification Criteria</td>
<td>Criteria formed from specific parameters and the values that they must contain to match devices that Discovery finds in the network with CIs in the CMDB. For a detailed discussion of these parameters, see <a href="#">Discovery classification parameters</a>.</td>
</tr>
<tr>
<td>SNMP OID Classifications</td>
<td>Unique “fingerprints” of all the SNMP devices that ship with the base Discovery product. Users can add OIDs for SNMP devices not in this list. This related list is available only for SNMP devices.</td>
</tr>
<tr>
<td>Triggers probes</td>
<td>Exploration probes that Discovery launches to gather detailed information about a CI that it has classified in the network. If you want to use patterns for horizontal discovery, add the <strong>Horizontal Pattern</strong> probe in the <strong>Probe</strong> column, and then specify your pattern in the <strong>Pattern</strong> column.</td>
</tr>
</tbody>
</table>

**Warning:** Do not specify your pattern in **Probe** column. Choose the Horizontal Pattern probe, which launches the specified pattern.
<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Versions</td>
<td>Lists versions of this classifier. A new version is created whenever you modify the classifier record. To revert to a previous version, open that record and select <a href="#">Revert to this version</a> under <a href="#">Related Links</a>.</td>
</tr>
</tbody>
</table>

This example shows a completed CI classification form with exploration probes defined. For instruction on creating probes, see [Discovery probes and sensors](#). The probes defined here are launched when the device is properly classified, unless Discovery is configured to stop after classification.
SNMP Classification
F5 BIG-IP Load Balancer

After the Sensor processes the results of the SNMP - Classify Probe, secondary to SNMP scanning, the On classification script gets run.

- **Name**: F5 BIG-IP Load Balancer
- **Active**: checked
- **Order**: 500

On classification script:

```
1 // This script gets run when something
```

Update Delete

Classification Criteria
SNMP OID Classifications (34)
Triggers probes (3) Versions

Classifiers:
- **F5 BIG-IP Load Balancer**

Probe:
- Name

MultiProbe-SNMP - F5 BIG-IP - Identity
SNMP - F5 BIG-IP - Identity
Create a Discovery process classification

A process classification allows Discovery to create a particular CI type from information gathered during the identification and exploration phases.

When a process matches the classification criteria, Discovery uses the process classification record to create a CI. You can also have Discovery update existing CIs or ignore certain processes by creating a process handler.

1. Navigate to Discovery Definition > CI Classification > Processes.
2. Click New.
3. Enter the classification fields (see table).

### Classification fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>Select the table where this classification generates CI records. This table must be an extension of the Applications (cmdb_ci_appl) table.</td>
</tr>
</tbody>
</table>
| Relation type | Select the CI relationship type for this classification. The relationship field is only available for Process and Scan Application classifications. Discovery process classifications typically use one of these relationship types:
  - Runs on::Runs: Defines the relationship of an application to the host on which it runs. This relationship is expressed from the perspective of the host and the application. For example: My database application runs on server001::server001 runs my database application.
  - Depends on::Used by: Defines the relationship of an application that communicates with another application. This relationship is expressed from the perspective of each application. For example: The Tomcat application depends on the MySQL database:: The MySQL database is used by Tomcat.
  - Virtualized by::Virtualizes: Defines the relationship of a virtual machine to its host. This relationship is expressed from the perspective of the virtual machine and of the host. For example: server001 is virtualized by Server ESX::Server ESX virtualizes server001. |
<p>| Active        | Select this option to enable the process classification record. Only active process classifications can create application CI records. |
| Order         | Enter the order in which Discovery should run this process classification when there are multiple classifications available for a table. Discovery runs process classifications from the lowest to highest order. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test with</td>
<td>Lists the host CI where an automatically generated process classification conditions were met. Use this field to test changes to the process classification to ensure that the updated classification behaves as expected.</td>
</tr>
<tr>
<td>Condition</td>
<td>Use the condition builder to create the match and classification criteria for the process classification. This field replaces both the Match criteria field and Classification Criteria related list. The upgrade process converts all existing classification criteria into conditions.</td>
</tr>
<tr>
<td>Note:</td>
<td>Condition filters in process classifiers are case-sensitive.</td>
</tr>
</tbody>
</table>
| On classification script     | Enter a script to run when the condition and classification criteria are met. Use this script to perform any special tasks after a device is classified. It is possible to use the g_probe_parameters hashmap from within a classification script to set probe parameters for any configured, triggered probes. For example, this code sets a 'node_port' parameter to 16001 for all triggered probes.  
  
  ```javascript
  g_probe_parameters['node_port'] = 16001;
  ```
  
  See On classification script objects for Discovery for more examples and for a list of the objects you can use in this kind of script. |
<p>| Triggers probes              | Select the exploration probes you want Discovery to launch. These probes gather detailed information about a classified CI. Discovery will not launch these probes when it is configured to stop after classification. If you want to use patterns for horizontal discovery, add the Horizontal Pattern probe in the Probe column, and then specify your pattern in the Pattern column. |
| Warning:                     | Do not specify your pattern in Probe column. You must chose the Horizontal Pattern probe, which launches the specified pattern.                                                                              |
| Applications                 | Use this related list to view the application CIs that match this process classification.                                                                                                                      |
| Test results                 | Use this related list to view the how Discovery classifies processes on the Test with host and build better classification conditions.                                                                          |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td>Use this related list to view the parameters associated with this process and build better classification conditions. See Discovery classification parameters for a list of the parameters you can use.</td>
</tr>
<tr>
<td>Versions</td>
<td>Use this related list to view previous versions of the process classification record.</td>
</tr>
</tbody>
</table>

4. Right-click the header bar and click Save.
5. Enter items from related lists.
6. Click Update.
### Process Classification

**New York SQL Temp**

<table>
<thead>
<tr>
<th>Name</th>
<th>MYCustomSQLServer2014</th>
</tr>
</thead>
</table>

**Active:** True

**Order:** 100

**Condition:**
- **Add Filter Condition:** MYCustomSQLServer2014
- **Operator:** And
- **Value:**

```
// This script gets run when something gets classified
```

**Related Links**

Create a "new class" process handler.

**Triggers probes**

<table>
<thead>
<tr>
<th>Name</th>
<th>MYCustomSQLServer2014</th>
</tr>
</thead>
</table>

No records to display
Run a discovery through the **Discovery Schedule** to search for applications. You can also **create or modify process handlers** that prevent the creation of duplicate CIs based on process classifiers if that is an issue for your environment.  

**On classification script objects for Discovery**  
Use an **On classification** script in a process classifier to customize an application record. This kind of script is used in a process classifiers.

---

**Renaming the default application name**

By default, application names are in this format: `<name of the process classifier>@<the name of the computer CI where the process resides>`;

For example, for a MySQL server running on a computer called `machineA`, the application is named `mysql@machineA`.

You can use the **On classification script** field in the **process classifier record** to change the default application name to match your business needs. For example, the following script changes the default application name to include a suffix after the process classifier:

```javascript
var computerName = g_sensor.deviceGR.name;
var processClassifierName = g_classification.name;
current.name = processClassifierName + "999" + "@" + computerName;
```

In this example, the name of the application record becomes `mysql999@machineA`.

Another common technique is to set the application name based on the name, command, and parameter variables. For example, an Eclipse process might have the following values in these variables:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>&quot;eclipse&quot;</td>
</tr>
<tr>
<td>command</td>
<td>&quot;/glide/eclipse/Eclipse.app/Contents/MacOS/eclipse&quot;</td>
</tr>
<tr>
<td>parameter</td>
<td>&quot;-psn_0_1884620&quot;</td>
</tr>
</tbody>
</table>

If an Eclipse application runs on a computer called `machineA`, ServiceNow names the application `eclipse@machineA`. The following script appends the parameter value as part of the application name.

```javascript
var computerName = g_sensor.deviceGR.name;
var processClassifierName = g_classification.name;
current.name = processClassifierName + parameters + "@" + computerName;
```

In this example, the name of the application record becomes `eclipse-psn_0_1884620@machineA`.

Sometimes it is useful to pass values to the triggered probes in the process classification. You can do this by creating a custom script that defines a name/value pair for the `g_probe_parameters` object. For example:

```javascript
g_probe_parameters['processCommand'] = command;
```

In this example, when a classification record triggers a probe, the script passes the probe a parameter called `processCommand` with the value of the `command` variable.
### Script objects

Use these objects in the script:

<table>
<thead>
<tr>
<th>Script object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>current</td>
<td>Points to a JavaScript object with its (property:value) pair to update the application record. (It is not an actual GlideRecord object of the application.)</td>
</tr>
<tr>
<td>g_sensor</td>
<td>Points to the running process sensor class. This object contains a deviceGR object, which points to the computer CI record on which the process resides.</td>
</tr>
<tr>
<td>g_classification</td>
<td>Points to the process classifier record itself.</td>
</tr>
<tr>
<td>name</td>
<td>Points to the process name.</td>
</tr>
<tr>
<td>command</td>
<td>Points to the process command.</td>
</tr>
<tr>
<td>parameters</td>
<td>Points to the process parameters.</td>
</tr>
<tr>
<td>g_probe_parameters</td>
<td>A JavaScript object that will allow parameter passing to the triggered probes.</td>
</tr>
</tbody>
</table>

Create a Discovery process handler

Process handlers prevent the creation of duplicate CIs by filtering out parameters known to have inconsistent values before process classification occurs. You can create new classifiers or edit existing ones.

1. Navigate to **Discovery Definition > CI Classification > Process Handlers.**
2. Click **New.**
3. Complete the fields on the Process Handler form.

#### Process Handler form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique name for the process handler record.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to run the process handler.</td>
</tr>
<tr>
<td>Classify</td>
<td>• Select the check box to enable classification of any Running Process (cmdb_running_process) record that matches this Process Handler's conditions.</td>
</tr>
<tr>
<td></td>
<td>• Clear the checkbox to disable classification of any Running Process (cmdb_running_process) record that matches this Process Handler's conditions.</td>
</tr>
<tr>
<td>Condition</td>
<td>Select the conditions that trigger the process handler to run the script. In most cases, this condition contains either specific executable names or the presence of certain parameters.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Script</td>
<td>Enter the JavaScript you want to run on the current record in the Running Process (cmdb_running_process) table when the Condition is met. The current variable is a reference to a Running Process GlideRecord. The script should examine current.parameters for certain values, perform string replacement to manipulate these values, and save the result to current.key_parameters. ServiceNow uses the key_parameters field, together with the process name, to determine whether the process is unique on a specific machine. See On classification script objects for Discovery for additional options.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

Run a discovery through the Discovery Schedule to search for applications and then verify the results in the CMDB for the applications that are found.

**Create an HTTP classification**

An HTTP classifier enables the horizontal discovery process to find devices via the HTTP protocol.

**Role required:** discovery_admin

You can create additional HTTP classifications if you want to classify devices via HTTP/REST, rather than by another protocol. Use HTTP classification for device discovery only, not for application discovery.

**Note:** By default, the **F5 HTTP classifier** is provided to classify F5 load balancers using REST. The **F5 LB REST** pattern is also provided to explore the device. You do not need to create a new HTTP classifier for F5 load balancers.

Of all protocols that Discovery uses (including WMI, SSH, and SNMP), HTTP is the lowest priority by default. Discovery uses HTTP classification only if:

- Shazzam determines that the ports for HTTP (80) and HTTPS (443) traffic are open.
- The horizontal discovery process fails for the higher priority port probes (WMI, SSH, and SNMP) if the ports for those protocols are not open, or if discovery for those protocols fail. The horizontal discovery process can fail, for example, if SSH and SNMP credentials are not configured or are incorrect.

**Note:** See **Port Probes** for more details on how port scanning works and to see the priorities of the different protocols.

HTTP classification launches the **HTTP Classify** probe to classify the device. The **HTTP - Classify** probe runs a GET request for each device for each HTTP classification. The URL of the request is built as follows: `PROTOCOL://IP:PORT/PATH`, where:

- **PROTOCOL** is port 80 for HTTP or port 443 for HTTPS.
- **IP:PORT** is the IP address and port number of the device.
- **PATH** is the path that you specify in the HTTP classifier.

1. Navigate to **Discovery Definition > CI Classification > HTTP**.
2. Click New.
3. Fill out the form fields (see table):

The F5 HTTP classification

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the classifier.</td>
</tr>
<tr>
<td>Path</td>
<td>Enter the path for the HTTP GET request.</td>
</tr>
<tr>
<td>Headers</td>
<td>Add an option header with a value. You can use a variable in the format <code>${variable}</code> instead of a static value. Supported headers depend on the REST web service provider you want to connect to. See the documentation for your web service provider to identify which headers are valid or required.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Port (IP Service) | If you want this classifier to use a custom port that is not already a default value in the IP Service (cmdb_ip_service) table, do the following:
1. Click the lookup icon to see the list of IP Services.
2. Click New and create a new port-to-protocol mapping.
3. Select that port-to-protocol mapping for this Port field.
Many commonly used protocols are already mapped to ports by default, such as http to port 80 and https to port 443. If you want to use one of these default port-to-protocol mappings, do not enter any values in the Port field. The HTTP port probe uses the correct one by default.

Order | Enter a number that represents the priority of this classifier. If Discovery finds more than one classifier that applies, it uses the classifier with lower Order number.

Protocol (optional) | If you configure a unique port-to-protocol mapping for the Port field, enter an optional protocol override that Discovery uses with this classifier. For example, if you want to use port 5000 and the HTTP protocol, add the 5000/HTTP port-to-protocol mapping to the Port field, and then choose http in the Protocol field.
This field is not visible by default. You must configure the form to add it.

4. Right-click the header and select **Save**.
A record is created in the HTTP Classifications (discovery_classy_http) table.
The HTTP Classification Match related list appears. This related list shows criteria that you can configure to match against the body of the response to the GET Request that Discovery sends.

5. In the HTTP Classification Match related list, define the criteria and specify the pattern to launch for this classifier. The criteria must be met for Discovery to use this HTTP classifier.
a) Click **New**.
b) On the HTTP Classification Match form, fill in the form fields (see table).
c) Click Submit.

d) If there are different patterns for different versions of the device you are trying to discover, define as many HTTP Classification Matches as necessary.

**Classification for IP address discovery**

Discovery provides a way to classify devices it finds through IP address discovery, even when no credentials are available.

When you run a discovery for IP addresses, as opposed to a CI discovery, the Discovery application makes certain assumptions about devices and the applications running on those devices from the ports that it finds open. Classification parameters for this type of Discovery are generated differently from scans in which credentials are available.
The syntax for creating parameters is derived from the fields returned by the Shazzam probe when conducting a Discovery for IP addresses. Parameters for CIs and applications are formed in the same way. The Shazzam probe creates an XML file containing the following fields:

- name
- port
- portprobe
- protocol
- result
- service

**Note:** Optional fields that can be used to form parameters appear as child tags beneath the default fields. Example of these are the `sysDescr` and `banner_text` fields.

Parameters are expressed in the form of `<portprobe.service.field>`. The value for field can come from any of the fields or child tags in the XML file. For example, the following parameters classify a device as a UNIX server and detect an installation of MySQL:

- ssh.ssh.result
- mysql.mysql.result

These parameters were derived from the values in the following XML file generated by a Shazzam probe conducting an IP Scan. The result field returned a value of open for ports 22 and 3306 on the target device. The service field indicates the services that normally communicate over those ports.

The `sysDescr` field can provide additional information about devices, depending upon the manufacturer. This XML file from the Shazzam probe reveals the following about port 161 on the device at IP 10.10.11.149:
In the classification criteria, we can construct the following parameter with `sysDescr` that returns an Apple AirPort wireless router:

```
<sysDescr>
Apple AirPort
<sysDescr>
```

Modify classifiers for IP address discovery

When you run an IP address type of discovery, port probes scan devices without the use of credentials, and then Discovery can determine which classifiers to use. You can add port probes and additional classifiers for IP address discovery.

Role required: admin

Although no credentials are required to scan for Windows or UNIX devices with this type of scan, credentials are still required for SNMP devices. By determining which ports are open on the devices that it scans, IP address classification can discover such things as the type of device (computer, UPS, etc.), operating system, running applications, and version numbers.

**Note:** IP address classification attempts to classify devices when no credentials are available; however, Discovery will use credentials when they are available, even when IP address classification is configured.

To use IP address classification, follow these steps:

1. Determine what ports to use for classification. Run a scan program such as Nmap on specific IP addresses to decide which ports reveal the desired information about a device or application.

   The scan can reveal several pieces of data that are useful for configuring classification parameters. An Nmap scan displays port numbers, their state (open or closed), their service names, and any version information it can find. From the port information returned in the example below, we can construct criteria to classify UNIX servers (port 22), MySQL (port 3306), and Apache Tomcat (port 16000).
2. Add an IP Service and port probe.

The out-of-box system supplies probes for some of the most common ports, but additional port probes will be needed for effective IP address scanning.

1. Navigate to Discovery Definition > IP Services and click New.

2. Create a new IP Service record using the port number and service from the Nmap scan. In this example, we associate the mysql service with port 3306 and add the CI (sanops02) on which the service runs to the Available on Related List.
3. To use Basic Discovery, navigate to **Discovery Definition > Functionality Definition** and select the record for **All**.

4. Add the new port probes to the list. This tells Discovery which port probes to run for IP address scans.
5. Save the record and navigate to Discovery Definition > Port Probes and click New.
6. Create a port probe using the new IP Service you just defined.
3. Create a new classification and add the parameter for IP address scanning.

In this example, we have created an application classifier that will discover Apache Tomcat, based on the port information we received from the Nmap scan. See the following section for details about forming parameters for IP address scans.
4. Optional: In the Classification Criteria related list, create a criteria filter that determines when this classifier applies to the discovered devices. See the IP address classification parameters for a list of the parameters you can use.

Run a IP address discovery through the Discovery Schedule to search for devices.

Use Windows Remote Management for classification
You can configure the discovery of Windows hosts using the Windows Remote Management (WinRM) protocol.

- Install and configuration a Windows MID Server on the local network.
- WinRM service must be enabled on all discoverable Windows hosts.

Roles required: discovery_admin, agent_admin, admin

By default, the system uses the WMI protocol for device classification of Windows hosts. Administrators can instead use the WinRM protocol for more efficient lightweight data transfer and remote command execution.

1. Enable the WinRM service on all Windows hosts you want to discover.
2. Navigate to Discovery > MID Servers.
3. Select the MID Server you will use for discovery of Windows hosts.
   The system displays the MID Server record.
4. From the Configuration Parameters related list click New.
   The system displays a MID Server Configuration Parameter record.
5. In the Parameter name field, select the mid.windows.management_protocol parameter from the choice list.
6. Enter a value of WinRM.

7. Click Submit.
   The system displays the MID Server record.
8. Optional: Add other Windows Remote Management protocol parameters as needed.
### Additional WinRM parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default</th>
<th>Requires MID Server restart</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid.powershell_api.winrm.remote_port</td>
<td>Specifies the communications port the MID Server uses to communicate with the WinRM protocol.</td>
<td>5985</td>
<td>No</td>
</tr>
<tr>
<td>mid.powershell_api.session_pool.target.max_size</td>
<td>Specifies the maximum number of sessions allowed in the pool per target host.</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>mid.powershell_api.session_pool.max_size</td>
<td>Specifies the maximum number of sessions allowed in the session pool.</td>
<td>100</td>
<td>Yes</td>
</tr>
<tr>
<td>mid.powershell_api.idle_session_timeout</td>
<td>Specifies the timeout value of idle Powershell sessions in seconds.</td>
<td>60</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Run a discovery from the [Discovery schedule](#) to find Windows machines on your network.  

**Reclassify a Windows Workstation machine as a server**

By default, Discovery automatically classifies computers using certain Windows operating systems as workstations. However, you might want specific computers in your network that are acting as servers to be classified by their function and not their operating system.

Use the following variables, preceded by `cidata`, to construct a reclassification condition. For example, to reclassify based on a machine's IP address, use `cidata.ip_address`.

- name
- dsn_domain
- os_domain
- ip_address
- serial_number

The following procedure reclassifies any Windows workstation operating system (Windows Vista, XP, or Windows 7) that is acting as a server.

1. Navigate to **Discovery Definition > CI Classification > Windows**.
2. Create a new classification record, such as Windows XP Server.
4. Right-click in the header bar and select **Save** from the context menu.  
   The **Classification Criteria** and **Triggers Probes** Related Lists appear.
5. Configure the following **Classification Criteria**:
Select a variable to use as the classification criteria from the list above. For example, to reclassify a machine by name, enter `cidata.name`. This works for servers that have a uniform naming convention, such as SRV001, SRV002, etc., regardless of operating system.

Select an operator for the classification condition. In networks containing servers named with a specific convention, you might select `starts with` or `contains`.

Enter the value for the condition. In our example of a network with a server naming convention, this value would be the root of that convention, such as `SRV`. This condition will classify all computers as servers if their machine name is SRVXXX.

6. Select the **Triggers Probe** related list and add the appropriate probes.

   a) Copy the list of probes from another Windows server classification, including the **Condition scripts**.

   b) Ensure that the **Windows - Identity** probe has a phase of **Identification** (the default is **Exploration**).

   The completed form looks like this:
Run a discovery from the Discovery schedule to find Windows machines on your network, and then check the cmdb_ci_win_server table and related tables to see how data is populated in the CMDB.
Discovery classification parameters
These parameters are available for Discovery classifiers.

Note: Condition filters in process classifiers are case-sensitive.

Unix parameters

The UNIX parameters define the characteristics of several types of computers, such as Linux, Solaris, and HP-UX, communicating with SSH protocol, version 2.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>output</td>
<td>The raw output of the classifier probe (uname -a).</td>
</tr>
<tr>
<td>type</td>
<td>Returns the classification UNIX.</td>
</tr>
<tr>
<td>IP address</td>
<td>Returns the IP address of the device being discovered.</td>
</tr>
<tr>
<td>name</td>
<td>Name of the operating system for this UNIX CI. For example, Linux or HP-UX.</td>
</tr>
</tbody>
</table>

Windows parameters

Windows parameters identify Windows computers communicating with the WMI protocol.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>isNode</td>
<td>Indicates if this instance is a node.</td>
</tr>
<tr>
<td>type</td>
<td>Returns the classification windows.</td>
</tr>
<tr>
<td>isVIP</td>
<td>Indicates if this CI is a virtual machine, with a virtual IP address.</td>
</tr>
<tr>
<td>ip_address</td>
<td>Returns the IP address of the device being discovered.</td>
</tr>
<tr>
<td>name</td>
<td>Name of the Windows version.</td>
</tr>
</tbody>
</table>

SNMP parameters

The SNMP parameters can define the characteristics of several types of devices, such as routers, switches, and printers.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>powering</td>
<td>A value of true indicates that this device is an uninterruptible power supply (UPS).</td>
</tr>
<tr>
<td>hosting</td>
<td>A value of true indicates that this device can host programs. Hosts are general purpose computers such as servers.</td>
</tr>
<tr>
<td>netware</td>
<td>A value of true indicates that this device is running the Netware operating system.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>routing</td>
<td>A value of true indicates that this device has network routing capabilities.</td>
</tr>
<tr>
<td>ip_address</td>
<td>Returns the IP address through which the device is being discovered. A device can have multiple IP addresses.</td>
</tr>
<tr>
<td>sysdescr</td>
<td>Required descriptive field on any SNMP device that can contain useful classification data, such as the operating system and its version.</td>
</tr>
<tr>
<td>vlans</td>
<td>A value of true indicates that this device can host a virtual local area network.</td>
</tr>
<tr>
<td>hint_router</td>
<td>A value of true indicates that Discovery has determined that this device is a router. This field only applies to devices that can be used as both a router and a switch.</td>
</tr>
<tr>
<td>block_router_exploration</td>
<td>If this parameter is true, Discovery will not launch exploration probes for routers it detects. This parameter is used for network Discovery only.</td>
</tr>
<tr>
<td>switching</td>
<td>A value of true indicates that this device has network switching capabilities.</td>
</tr>
<tr>
<td>mfr_apc</td>
<td>A value of true indicates that this device is an uninterruptible power supply (UPS) manufactured by American Power Conversion (APC).</td>
</tr>
<tr>
<td>printing</td>
<td>A value of true indicates that this device has printing capabilities.</td>
</tr>
<tr>
<td>block_switch_exploration</td>
<td>If this parameter is true, Discovery will not launch exploration probes for switches it detects. This parameter is used for network Discovery only.</td>
</tr>
</tbody>
</table>

**Process parameters**

Process parameters identify processes such as those used by LDAP, Apache Server, and JBoss Server.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameter</td>
<td>The parameters used to run the process.</td>
</tr>
<tr>
<td>command</td>
<td>The command used to run the process.</td>
</tr>
<tr>
<td>output</td>
<td>The complete output of the current line of the process probe.</td>
</tr>
<tr>
<td>type</td>
<td>Indicates the process type (e.g. unix or windows).</td>
</tr>
<tr>
<td>PID</td>
<td>The process ID generated by the operating system of a device to identify a running process. Generally, this parameter is not a practical classification criteria, because the value does not remain static, except in the case of processes running on an appliance that is never restarted.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>name</td>
<td>Name of the process being discovered. In some cases, this parameter is not reliable, since several processes might be given the same name. In Windows, for example several processes return scvhost.exe for this parameter.</td>
</tr>
</tbody>
</table>

**Check IP service affinity for Discovery and Orchestration**

You can check the IP Services table for a list of IP addresses that are associated with a protocol.

**Role required:** admin

The IP Services table maps a port to a protocol. Several mappings are provided by default for commonly used port-protocol combinations, such as port 80 for HTTP, port 22 for SSH, and port 161 for SNMP.

A system property called `glide.discovery.ip_service_affinity` allows Discovery to remember the last port of the IP address that was discovered. This property is set to `false` by default.

**Caution:** You should not modify IP services unless your organization uses custom ports.

1. Navigate to **Discovery Definition > IP Services**.
2. Filter the list to find the appropriate IP service.
3. Click the name of the service to go to that IP service page.
4. Click the **IP Service Affinities** tab for the list of IP addresses associated with that service.
Discovery identifiers

After Discovery classifies a CI, it uses identifiers to determine if the device already exists in the CMDB.

Discovery launches special identity probes that accumulate identification data for each device and feed that data into the identifiers, which determine the action that Discovery must take for each device. Identifiers accurately determine the identity of the device to prevent the creation of duplicate CIs. This identification step only takes place for the Configuration item type of discovery, not for the other types of discovery.

The identity probe in the base Discovery system can be configured to ask the device for information such as its serial numbers, name, and network identification. The results of this scan are processed by an identity sensor, which then passes the results to the identifier. The identifier then attempts to find a matching device in the CMDB. If the identifier finds a matching CI, the identifier either updates that CI or does nothing. If the identifier cannot find a matching CI, it either creates a new CI or does nothing. If Discovery is configured to continue, the identifier launches the exploration probes configured in the classification record to gather additional information about the device. Exploration probes can be multiprobes or simple probes.

**Important:** Serial numbers are necessary for accurate asset tracking. If you modified baseline probes, sensors, or patterns, verify that they still discover serial numbers. In addition, do not configure sensors or patterns to modify the serial number syntax, such as adding a custom prefix. Non-standard serial numbers can lead to inaccurate asset tracking.

This diagram shows the processing flow for classifying and probing devices with identifiers configured.
CMDB identifier tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifier (cmdb_identifier)</td>
<td>Stores all identifier rules.</td>
</tr>
</tbody>
</table>
## Identifier rules

The default Discovery system contains these identifier rules, each of which is associated with a specific CI type (the `sys_class_name` field on the CI record) or the table in the **Applies to** field and contains the appropriate attributes for discovering CIs from the specified table. Where necessary to discover all possible occurrences of an attribute, tables from related lists (Search on tables) are included in the rule. For more information, see [Create or edit a CI identification rule](#).

### CMDB identifier rules

<table>
<thead>
<tr>
<th>Rule</th>
<th>Applies to table/attributes</th>
<th>Search on table/attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESX Server Rule</td>
<td>ESX Server (<code>cmdb_ci_esx_server</code>)</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>· correlation_id</td>
<td></td>
</tr>
<tr>
<td>Hardware Rule</td>
<td>Hardware (<code>cmdb_ci_hardware</code>)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· serial_number</td>
<td>Serial Number (<code>cmdb_serial_number</code>)</td>
</tr>
<tr>
<td></td>
<td>· serial_number_type</td>
<td>· serial_number_type</td>
</tr>
<tr>
<td></td>
<td>· name</td>
<td>· name</td>
</tr>
<tr>
<td></td>
<td>· ip_address</td>
<td>· ip_address</td>
</tr>
<tr>
<td></td>
<td>· mac_address</td>
<td>· mac_address</td>
</tr>
<tr>
<td>Storage Server Rule</td>
<td>Storage Server (<code>cmdb_ci_storage_server</code>)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· cim_object_path</td>
<td>Serial Number (<code>cmdb_serial_number</code>)</td>
</tr>
<tr>
<td></td>
<td>· name</td>
<td>· serial_number</td>
</tr>
<tr>
<td></td>
<td>· serial_number</td>
<td>· serial_number_type</td>
</tr>
<tr>
<td></td>
<td>· serial_number_type</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· mac_address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· ip_address</td>
<td></td>
</tr>
<tr>
<td>WBEM Service Rule</td>
<td>WBEM Service (<code>cmdb_ci_wbem_service</code>)</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>· cim_object_path</td>
<td></td>
</tr>
</tbody>
</table>

### Matching strategy for the hardware rule

The `sys_class_name` cannot be an attribute for independent rules, such as `cmdb_ci_hardware`. If your Discovery identification strategy depends on matching a CI with a specific class, you must create a new rule for each class you want to use for matching and specify that class in the **Applies to** field of the Identifier form.
For example, you can create an identifier for a Linux server with different attributes than the Hardware Rule. You might want to use the machine name, IP address, and MAC address for identification. This is a solution for networks that use NIC bonding or teaming to increase available bandwidth. Bonded interfaces appear to be the same physical device and share the same IP and MAC addresses. The use of the name attribute allows Discovery to differentiate between the individual interfaces in the bonded channel.

**Caution:** If you create an identifier with the name attribute, avoid changing adapter names. Discovery will be unable to resolve existing CIs for renamed adapters. Discovery labels the Install Status of that CI as “Absent” and creates another CI.

Your new rule would look like this:
### Linux identifier rule

**Name:** Linux Rule  
**Active:** ✓  
**Applies to:** Linux Server (cmdb_ci_linux_server)  
**Description:** Identifier for Linux servers  
**Independent:** ✓

<table>
<thead>
<tr>
<th>Identifier Entries</th>
<th>Go to</th>
<th>Priority</th>
<th>Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifier = cmdb_ci_linux_server</td>
<td></td>
<td>Search on table</td>
<td>Criterion attributes</td>
</tr>
<tr>
<td>✓</td>
<td>Active</td>
<td>Linux Server (cmdb_ci_linux_server)</td>
<td>name</td>
</tr>
<tr>
<td>✓</td>
<td>Active</td>
<td>Linux Server (cmdb_ci_linux_server)</td>
<td>ip_address</td>
</tr>
<tr>
<td>✓</td>
<td>Active</td>
<td>Network Adapter (cmdb_ci_network_adapter)</td>
<td>mac_address</td>
</tr>
</tbody>
</table>

Actions on selected rows...
Evaluation order for Discovery identifiers

Custom identifiers must have different Order values than those of the default identifiers. Discovery parses identifiers and attributes in sequence from low order numbers to high. You can create identifiers to run before or after the default identifiers, or mixed in with the identifiers from the base system. To prevent any identifier or rule from running, disable it by clearing the Active check box. The evaluation order for CMDB identifiers is established within each rule and only controls the parsing order of the attributes in that rule.
### Identifier Table

<table>
<thead>
<tr>
<th>Active</th>
<th>Allow fallback to parent's rule</th>
<th>Allow null attribute</th>
<th>Criterion attributes</th>
<th>Optional condition</th>
<th>Enforce exact count match (Lax mode)</th>
<th>Priority</th>
<th>Search on table</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>True</td>
<td>False</td>
<td>comigration_id</td>
<td>false</td>
<td>false</td>
<td>75</td>
<td>Hardware</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
<td>serial_number</td>
<td>false</td>
<td>true</td>
<td>333</td>
<td>Serial_number</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
<td>serial_number</td>
<td>true</td>
<td>true</td>
<td>376</td>
<td>Hardware</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
<td>name</td>
<td>True</td>
<td>true</td>
<td>376</td>
<td>Network_adapter</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
<td>ip_address</td>
<td>true</td>
<td>true</td>
<td>376</td>
<td>Network_adapter</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>False</td>
<td>mac_address</td>
<td>false</td>
<td>false</td>
<td>376</td>
<td>Network_adapter</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Properties for processing duplicate CIs

You can control how Discovery handles duplicate CIs with properties installed with Identification and Reconciliation. Use the glide.identification_engine.skip_duplicates and glide.identification_engine.skip_duplicates.threshold properties. For more information, see Properties for Identification and Reconciliation.

Properties that control identifier versions

All instances use identifiers from the CMDB Identification and Reconciliation framework. Upgrades from pre-Geneva versions still preserve the legacy identifiers, but you can switch to the new identifiers using a property: glide.discovery.use_cmdb_identifiers. If you upgraded from a pre-Geneva version, you must manually add this property and set it to true to use the new identifiers. If you upgraded from Geneva or later releases, this property is available in the System Properties (sys_properties) table. To preserve functionality in custom legacy identifiers, convert them to the new CMDB identifier rules format before enabling this property. The system does not reconfigure your custom identifiers to the new framework automatically.

Note: When Service Mapping is active, the new identifiers from the CMDB Identification and Reconciliation framework are always used regardless of the property value.

How Discovery identifiers work

When Discovery has determined the device’s class, it launches an identity probe that is configured to run one or more commands with a single authentication.

The identity probe in the base Discovery system can be configured to ask the device for information such as its serial numbers, name, and network identification. The results of this scan are processed by an identity sensor, which then passes the results to the identifier. The identifier then attempts to find a matching device in the CMDB. If the identifier finds a matching CI, the identifier either updates that CI or does nothing. If the identifier cannot find a matching CI, it either creates a new CI or does nothing. If Discovery is configured to continue, the identifier launches the exploration probes configured in the classification record to gather additional information about the device. Exploration probes can be multiprobes or simple probes.

Note: When you use patterns for Discovery, the identity probe is not used. Discovery uses the appropriate identifier rules based on the CI type that you are trying to discover as specified in the pattern operations. The operations in the pattern along with these identifier rules perform the identification and exploration phases of discovery.

This diagram shows the processing flow for classifying and probing devices with identifiers configured.
Configure Discovery identity probes

Identity probes are multi-probes, which contain one or more simple probes configured to extract specific information from manageable devices. You can create your own identity multi-probe to identify CIs that Discovery does not already identify.

Role required: discovery_admin, admin
Identity probes return such information as device serial numbers (there can be several per device), the computer name, and network identification (MAC address). For example, the Linux - Identity multiprobe provided in the system, contains two simple SSH probes configured to return hardware and network information about the device. After the probe has discovered this information, it passes the data to a multisensor configured to process identity information.

**Note:** A multiprobe cannot contain another multiprobe.

1. Navigate to **Discovery Definition > Probes**.
2. Select a probe from the list whose **Class** is Multiprobe.
3. Make sure the scope is set to **Global**.
4. In the **Includes Probes** Related List, click **Edit** to select simple probes to include in this multiprobe.
5. Click **New** to create a new simple probe.
Configure Discovery identity sensors that are part of the identify probes.

If you customize an identify multi-probe, you can also configure a multi-sensor for it.

Role required: discovery_admin, admin

These multi-sensors pass the data returned by the identity probes to the Discovery identifiers. The identifiers then search the CMDB for matching CIs. When the identity of a device is resolved, the identifiers pass the result state for the device to the multi-sensors, which react accordingly, either by launching exploration probes or stopping Discovery for that device.

1. Navigate to Discovery Definition > Sensors.
2. Select an existing identity multi-sensors from the list.

   **Note:** To avoid confusion, the multi-sensors provided in the instance must have the same names as their matching multi-probes.

The Responds to Probes related list shows the simple probes which pass their data to this multi-sensors. These are the same simple probes that appear in the Includes Probes list in the matching multi-probes record.

3. Click the Script link for each probe in the list to see the script that the multi-sensors runs to process the data from the probe.
Run a discovery through the Discovery Schedule to search for CIs and verify that they are identified correctly in the CMDB.

**Serial number types for identification**

As Discovery finds CIs, their serial numbers are listed in the Serial Number (cmdb_serial_number) table so they are easy to identify. Serial number types vary depending on the CI, as described in the following examples.

**Important:** Serial numbers are necessary for accurate asset tracking. If you modified baseline probes, sensors, or patterns, verify that they still discover serial numbers. In addition, do not configure sensors or patterns to modify the serial number syntax, such as adding a custom prefix. Non-standard serial numbers can lead to inaccurate asset tracking.

**Linux**

In the result of dmidecode | cat, the value on the left side is what Discovery looks for. The value on the right side is how it is stored in the Serial Number (cmdb_serial_number) table.

<table>
<thead>
<tr>
<th>Serial Number</th>
<th>Expected Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>system_serial_number</td>
<td>Serial Number : system_serial_number</td>
</tr>
<tr>
<td>uuid_serial</td>
<td>UUID : uuid_serial</td>
</tr>
<tr>
<td>baseboard_serial</td>
<td>Serial Number : baseboard_serial</td>
</tr>
<tr>
<td>chassis_serial</td>
<td>Serial Number : chassis_serial</td>
</tr>
</tbody>
</table>

**Windows**

For Win32 WMI classes, the value on the left is the name by which it is stored in the Serial Number (cmdb_serial_number) table. The value on the right is the WMI value.

```python
{'system' : Win32_ComputerSystemProduct.IdentifyingNumber};
{'uuid' : Win32_ComputerSystemProduct.UUID};
{'chassis' : Win32_SystemEnclosure.SerialNumber};
{'bios' : Win32_BIOS.SerialNumber};
{'baseboard' : Win32_BaseBoard.SerialNumber};
```

**SNMP**

For SNMP, the mapping below is based on the code. Physical types of serial numbers are from all instances of iso.org.dod.internet.mgmt.mib-2.entityMIB.

```python
'cisco_stack' :
    'iso.org.dod.internet.private.enterprises.cisco.workgroup.ciscoStackMIB.chassisGrp.chassisSerialNumberString'
'cisco_chassis' :
    'iso.org.dod.internet.private.enterprises.cisco.temporary.chassis.chassisId'
'foundry' :
    'iso.org.dod.internet.private.enterprises.foundry.products.switch.snChassis.snChasGen.snChasSerNum'
```
Discovery status

The Discovery status provides a summary of a Discovery launched from a schedule. You can also cancel a Discovery that is in progress from the status form.

To access the Discovery Status form, navigate to **Discovery > Status** and open the status record for a Discovery.

Each record in the Status list represents the execution of a Discovery by a schedule and displays such high level information as the date of the Discovery, the mode, the number of probe messages sent to devices, and the number of sensor records that were processed. A status record contains data that can help you troubleshoot a failed discovery. Use this data to troubleshoot the behavior of individual probes and sensors or even run those elements separately. Use the status controls to enter probe/sensor threads at any point for a specific Discovery, and then follow the process in either direction.

- **Refresh**: Refreshes the related tabs, including the **Discovery Log** and **ECC Queue** for discoveries still in progress.
- **Discovery timeline**: The Discovery timeline is a graphical display of a discovery, including information about each probe and sensor that was used.
- **Discovery log entries**: The Discovery Log shows information such as classification failures, CMDB updates, and authentication failures. A Discovery Log record is created for each action associated with a discovery status.
- **ECC queue entries**: Entries in the ECC queue provide you with a connected flow of probe and sensor activity, as well as the actual XML payload that is sent to or from an instance.
- **Device history**: The device history provides a summary of all the devices scanned during discovery, and what action sensors took on the CMDB.

**Note**: By default, only 30 days of Discovery records are displayed in the status list at a time.

The following Discovery status record shows the discovery of a load balancer using a pattern.
The Discovery Status contains the details of a Discovery that has been launched from a Discovery Schedule. The Status displays the status of the Discovery and the ECC queue. Read more about the Discovery Status or find assistance with Discovery troubleshooting.

Number: DIS0010042
Description: Discover Now
Schedule: load balancer-f5

From Schedule
Discover: Configuration items
Max run time: 00:00:00

Update  Delete

Related Links
Refresh
Show Discovery timeline

Discovery Log (5) Devices (1) ECC Queue (35)

Discovery Log

Go to Created  Search

Discovery Log

2017-03-31 15:28:18  Information  Discovery completed

Example Discovery status record
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>An auto-generated number for the record.</td>
</tr>
<tr>
<td>Description</td>
<td>How this Discovery was run. Typically, the description is Scheduled, but if you ran Discovery manually from a Schedule, the record would show Discover Now in the Description field. If the scan was performed by the Help the Help Desk application, this value is Help the help desk.</td>
</tr>
<tr>
<td>Schedule</td>
<td>The name of the Discovery schedule.</td>
</tr>
<tr>
<td>State</td>
<td>The state of the discovery:</td>
</tr>
<tr>
<td></td>
<td>• Starting</td>
</tr>
<tr>
<td></td>
<td>• Active</td>
</tr>
<tr>
<td></td>
<td>• Completed</td>
</tr>
<tr>
<td></td>
<td>• Canceled</td>
</tr>
<tr>
<td>Started and Completed</td>
<td>The number of probes started and completed during discovery.</td>
</tr>
<tr>
<td>Source</td>
<td>Optional read-only field that shows the source of the Discovery, such as Schedule Discovery, or Quick Discovery. If the source of the Discovery was from an API call, the source is DiscoveryAPI.</td>
</tr>
<tr>
<td>From Schedule (Read-only schedule data)</td>
<td>Shows the Discovery type. The possible types are: Configuration items, IP addresses, Networks, or Web Service.</td>
</tr>
<tr>
<td>Discover</td>
<td>Shows the Discovery type. The possible types are: Configuration items, IP addresses, Networks, or Web Service.</td>
</tr>
<tr>
<td>Max Run Time</td>
<td>Displays the maximum amount of time Discovery was permitted to run on this schedule.</td>
</tr>
<tr>
<td>Include alive</td>
<td>Indicates that this Discovery includes devices on which one port responded to the scan, but no ports are open. Such a device is considered to be alive. If this check box is not selected, only active devices with one or more open ports that Discovery can query are displayed.</td>
</tr>
<tr>
<td>Log state changes</td>
<td>Indicates that state changes were logged during this Discovery. These states can be seen in the Last and Current fields in the Discovery Devices list in this form.</td>
</tr>
</tbody>
</table>

**Cancel a Discovery from a Discovery Status**
You can manually cancel a discovery while it is in progress from the Discovery Status record.
Role required: discovery_admin

1. Navigate to **Discovery > Status**.
2. Click a **Number** to open the status record.
3. Click **Cancel Discovery** in the **Related Links**.
   This action is only available for an active Discovery.

When a Discovery status cancels, any associated sensor transactions are immediately terminated and any scheduled sensor jobs are deleted from the system. After cancellation, the cleanup Status shows the **Completed count** and the cancellation is logged in the Discovery log. In the
Queue (ecc_queue) table, any records belonging to sensors exceeding the Transaction Quota Rule are set to the **Error** state.

**Discovery timelines**

A Discovery timeline generates a graphical display of a Discovery Status record, including information about each probe and sensor that was used in the discovery.

Use Discovery Timelines to display the following:

- The flow of **probes and sensors**, including those that are used with **patterns** through a discovery.
- The duration of each probe and sensor that ran during a discovery, and the proportion of time required for queuing and processing.
- Tooltips containing additional data about a probe and sensor.
- Records from the ECC Queue.

**Size of discoveries in the timeline**

View timelines for an entire discovery or for individual devices in a discovery. In the out-of-box system, the maximum size discovery that can be displayed in a timeline is 300 entries in the ECC Queue (150 probes and 150 sensors). To display larger discoveries, change the default setting in the `glide.discovery.timeline.max.entries` property.

**Example Discovery Timelines**

This example shows the Discovery timeline for the first two phases of Discovery (Port scan and Classification):

**Discovery DIS0010002 Timeline**

In this example, the Shazzam probe and sensor run, followed by the Unix classifier probe and sensor. The classifier calls the Horizontal Pattern probe, which runs a specific pattern.

**Discovery (early phases) on the timeline**

In this example, the Shazzam probe and sensor run, followed by the Unix classifier probe and sensor. The classifier calls the Horizontal Pattern probe, which runs a specific pattern.

View the Discovery timeline

A Discovery Timeline generates a graphical display of a Discovery Status record, including information about each probe, sensor, and pattern running.

Role required: admin, discovery_admin
1. Navigate to **Discovery > Status**.
2. Select a Discovery from the record list.
3. Click the **Show Discovery Timeline** Related Link.
4. Clear the warning, and then select the **Devices** Related List.
5. Click the IP address of a device.
6. In the Device record, click the **Show Discovery Timeline** Related Link.

   The Discovery timeline for that device appears.

   ![Discovery Timeline Diagram](image)

   **Discovery Timeline**

7. Use the pink slider at the bottom of the timeline to change the perspective.
   a) Move the slider from right to left to view all the tasks on a long timeline.
   b) Adjust the end points of the slider to change the magnification. A narrow slider zooms in on the spans and provides a more detailed view of complex timelines. A wide slider pulls the view out and makes more of the timeline visible on the screen.

8. Use the selector range at the top of the screen to adjust the visible time frame. To limit the timeline to the length of the Discovery, click **Max**.

   The time scale adjusts automatically to the length of the Discovery. The available time scale range is from one day to 1 year.

   **Tooltips**

   Discovery timelines display probe and sensor performance data and CI information in tooltips. Hover the cursor over a span to view this data. Probes are displayed by black spans. The queue time for a probe is shown as a silver bar within the span, and the processing time is represented by the remaining space. Sensor spans are red, and the queue time is shown as a green bar. Selected spans of any type display in yellow.
Discovery Timeline Shazzam Sensor

ECC Queue
Double-click a span to open the ECC Queue record for that probe or sensor.

Discovery Timeline ECC Queue
The ECC queue for Discovery
The External Communication Channel (ECC) Queue is a connection point between an instance and other systems that integrate with it, most commonly a MID Server.

Contents of the ECC Queue
The ECC queue contains records that are created during all stages of discovery. Each record is essentially a message from the instance and another system, classified as output, or a message from the external system to the instance, classified as input. ECC queue records provide you with a connected flow of probe and sensor activity, as well as the actual XML payload that is sent to or from an instance.

Automatic deletion of ECC queue records
If Discovery is running on an instance, the size of this table can grow to several gigabytes. Most of the accumulated data is unnecessary, but some entries might be important for troubleshooting any problems with Discovery. For example, if Discovery is not properly capturing the disk drives on a particular Windows server, look in the ECC Queue at the data returned by the Windows - System Information probe. You should retain ECC Queue data from Discovery for at least a month.

By default, records in the ECC Queue older than 7 days are deleted automatically. You can set the deletion schedule by updating the table rotation schedule for the ECC queue. The table rotation names are:

- ecc_queue_event
- ecc_queue

Accessing the ECC queue
You can access the ECC queue from either of these locations:

- Discovery > Output and Artifacts > ECC Queue
- ECC > Queue

The following image is an example of a record in the ECC queue. This record shows that a WMI classifier probe was instructed to run and has been processed. See MID Server ECC Queue for a description of each field.
An example ECC queue record

**ECC queue states**

Input and output messages keep processing in turn until the discovery is finished or terminated. You can monitor the ECC queue to see records populate during discovery, and see the states of each record change. The process is as follows:
1. When new probe instructions are available for the MID Server, the system creates an output record with the state of ready.

2. The MID Server then starts processing the probe instructions and the state changes to processing before it finally changes to processed.

3. When a result returns to the MID Server and is ready to be sent to the instance, the system creates an input record with the state of ready.

4. While the system processes the payload, the state changes to processing before it finally changes to processed.

   If any errors are encountered, the state changes to error. You can look for errors to troubleshoot failed discovery.

**Properties and parameters that affect the ECC queue**

These properties are available to control aspects of the ECC queue:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save ECC queue attachments</td>
<td>The normal behavior for discovery sensors is to delete attachments to ECC queue entries upon successful sensor processing. Enabling this property overrides this behavior, and forces attachments to be preserved. This would normally only be useful for debugging purposes. Location: Navigate to Discovery Definition &gt; Properties to enable the property.</td>
</tr>
</tbody>
</table>

**Discovery device history**

The Devices related list in the Discovery Status form provides a summary list of all the devices scanned.

During a Discovery, the list tracks current and completed activity and displays an incremental scan counter. When Discovery is finished for a device, the final disposition is displayed in the Completed activity column. Successful Discoveries that result in updated or created CIs are highlighted in green. To view the log entries for errors (such as connection failure) on a specific device, click the Details link in the Issues column.
Discovery device list

Click the IP address of a device in this list for details about that device. The log results for that device are displayed in the list at the bottom of the form.
The following fields are available in this form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>The IP address of the device discovered.</td>
</tr>
<tr>
<td>Completed activity</td>
<td>Indicates the outcome of Discovery for this device or the last completed activity for a Discovery in progress, such as Identified CI. Successful outcomes are indicated in green.</td>
</tr>
<tr>
<td>Current activity</td>
<td>The current scanning activity for this device for a Discovery in progress, such as Updating CI.</td>
</tr>
<tr>
<td>CMDB CI</td>
<td>The name of this device as it appears in the CMDB.</td>
</tr>
<tr>
<td>Started</td>
<td>The number of device-specific probes run. This number does not include the universal probes, Shazzam and Ping, that run initially.</td>
</tr>
<tr>
<td>Completed</td>
<td>The number of sensor records created from the device-specific probes that were run.</td>
</tr>
<tr>
<td>Scan status</td>
<td>Shows the final scan count of a completed Discovery or an incremental scan counter for a Discovery in progress. For example, Scan 17 of 19.</td>
</tr>
<tr>
<td>Issues</td>
<td>Displays the number of issues encountered during Discovery of this device. Select the Discovery Log Related List to view these issues.</td>
</tr>
<tr>
<td>Fields that can be added by configuring the form:</td>
<td></td>
</tr>
<tr>
<td>DNS Names</td>
<td>Displays DNS names for each discovered device.</td>
</tr>
</tbody>
</table>

If there were issues, or if Discovery failed to complete, click the Details link to view the log records for the issue. The failure of any probe is considered an issue, even if the device was eventually classified properly and updated in the CMDB.
Discovery details about a device

Address scan data

When Discovery scans for IP addresses only (without credentials or identifiers), no updates are made to the CMDB. All IP addresses discovered appear on this list, including multiple IPs on the same device. The results of IP address scans include slightly different information than the results of a CI scan. Since there is no CMDB activity associated with the IP address scan, the Completed activity column displays only the classification status.

Possible statuses are:
- Classified
- Unclassified
- Alive, not classified

For Classified devices, Discovery might identify the type of device in the Current activity column. For example, Network Gear might be classified as Cisco Network Gear, and a Computer might be classified as a Windows Computer.

Logs for Discovery
The Discovery log and Pattern Discovery/Horizontal Discovery log display the activity that takes place during a discovery. Use the logs to debug failed discoveries.

Discovery Log
The Discovery Log shows information such as classification failures, CMDB updates, and authentication failures. A Discovery Log record is created for each action associated with a discovery status.
Discovery Log Records

Pattern Discovery and Horizontal Discovery log

The Pattern Discovery log includes Horizontal Discovery log records, which display information about discoveries that were performed with patterns. A horizontal discovery log record is created for an entire horizontal discovery run, which includes the results of all the operations specified in the pattern.
View the Discovery Log
Open the Discovery Log to troubleshoot a discovery.

Role required: discovery_admin

1. Navigate to Discovery > Status.
2. Open a Discovery Status record.

Log entries appear in the Discovery Log related list.
### Discovery Log records

The Discovery Log provides the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created</td>
<td>Timestamp of the Discovery activity. Each timestamp defines the approximate time of the activity. Several Discovery events may occur in random order within a second.</td>
</tr>
<tr>
<td>Level</td>
<td>Classifies the activity into one of the following levels for general sorting:</td>
</tr>
<tr>
<td></td>
<td>- Error</td>
</tr>
<tr>
<td></td>
<td>- Information</td>
</tr>
<tr>
<td></td>
<td>- Warning</td>
</tr>
<tr>
<td>Short Message</td>
<td>Informative message detailing the outcome of the activity or the Discovery progress. Look here for the result of a classify probe or for authentication failure.</td>
</tr>
<tr>
<td>ECC queue input</td>
<td>The related input record from the ECC queue for this discovery. You can also view these records from the ECC Queue related list.</td>
</tr>
<tr>
<td>CI</td>
<td>Names a device for which a matching CI was found in the CMDB. Click the link to view the CI record for the device.</td>
</tr>
<tr>
<td>Source</td>
<td>Names the particular activity, such as the Shazzam probe or a UNIX classify probe.</td>
</tr>
</tbody>
</table>
**Field** | **Input value**
---|---
Device | Lists the IP address of the CI discovered. All devices identified by IP address appear in the log, even if they refused all invitations to communicate. Any port activity from a device places it into the log, even if all subsequent efforts to identify it fail. Click the IP address of the device to view the events associated with discovering that device.

3. Do any of the following:
- Click a link in the **CI column** to open the record for that CI.
- Click the link in the **Created** column to view the log record.
- Click the IP address link in the **Device** column to view the log records for a particular device.

**Note:** The Discovery log also displays errors that occur during identification by taking log entries from the Identification Engine log. See [Examine identification engine run logs](#) for more information about the identification engine and possible errors.

Open the Pattern Discovery Log
Open the Pattern Discovery Log to troubleshoot a discovery with patterns.

Role required: discovery_admin

1. Navigate to **Pattern Designer > Discovery Pattern Log**.
2. Sort the list of pattern log entries by any column, such as **Pattern** or **Status**.
3. Click the **View Log** link in the **Log message** column for the desired log record.
   The Horizontal Discovery Log window opens.
### The Horizontal Discovery Log

**Discovery Log Item** | **Description**
--- | ---
Pre Pattern Execution | The actions that were run before the pattern was launched.
Selecting Pattern for Execution | The pattern that was run for discovery.
{Identification Section name} | Displays the results of the operations in the pattern. Expand the name to see each operation.
Pre Payload Processing Scripts | The results of scripts that were run before the payload was received.
### Discovery Log Item

<table>
<thead>
<tr>
<th>Discovery Log Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payload Processing</td>
<td>Details about the payload and how it was processed. Look in this section to find errors that might have been encountered during various activities, such as the running of identification rules, updates to the CMDB, and so on.</td>
</tr>
<tr>
<td>Post Payload Processing Scripts</td>
<td>The results of scripts that were run after the payload was received.</td>
</tr>
</tbody>
</table>

4. Click an item in the left-hand column to see more information about it in the right-hand column.

5. To debug the pattern, click the name of the Identification Section, and then click the **Debug** button in the upper-right.

6. To return to the log record, click the **X** in the upper-right.

### Change Discovery log retention time

You can configure the amount of time that Discovery retains log entries.

**Role required:** admin

By default, the Discovery log retains information for 30 days, or 2,592,000 seconds.

1. Open the Auto Flushing table by typing `sys_auto_flush.list` in the navigation filter.
2. Open the **discovery_status** record.
3. Change the value in the **Age in seconds** field.
4. For the change to take effect immediately, run the Table Cleaner Job in the Schedule (`sys_trigger`) table.
5. Click **Update**.

### Patterns and horizontal discovery

A pattern is a series of operations that tell Discovery which CIs to find on your network, what credentials to use, and what tables to populate in the CMDB.

This topic assumes you understand the phases of horizontal discovery. If you need to review the horizontal discovery process for probes and for patterns, see the following:

- Horizontal discovery process flow with probes and sensors
- Horizontal discovery process flow with patterns

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

### How Discovery uses patterns

A pattern performs the same function as a probe: it identifies and explores a target CI. Discovery uses patterns only during the last two phases of discovery: Identification and Exploration.

When you kick off horizontal discovery with patterns, the Scanning and Classification phases run as they would if you were not using patterns (only using probes and sensors). After the Classification stage completes, Discovery looks at the trigger probes on the classifier to see which probe to launch. When the **Horizontal Pattern** probe is specified as a trigger probe, Discovery launches both the **Horizontal Pattern** probe and the pattern that it specifies.
Differences between probes and patterns

The main differences between using patterns for horizontal discovery and using only probes for horizontal discovery are outlined in this table:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Horizontal discovery with probes</th>
<th>Horizontal discovery with patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery phases</td>
<td>Probes run during all four discovery phases.</td>
<td>Patterns run only during the identification and exploration phases. Discovery still uses the Shazzam probe and port probes to kick off discovery.</td>
</tr>
<tr>
<td>Finding new CIs</td>
<td>Discovery offers several out-of-box probes and sensors. However, if you want to find new CIs or change the data that Discovery populates in the CMDB, you must create new probes and sensors or customize the existing ones. To do this, you need knowledge of Java and JavaScript, and you need to understand how probes work together with other probes and with sensors.</td>
<td>Discovery is offering more patterns in every major release and on the ServiceNow App Store. If you want to create a new pattern or customize an existing one, you can use the pattern designer, which is a user-friendly interface that helps you construct steps and enter values.</td>
</tr>
<tr>
<td>Feature support</td>
<td>Supports all standard network, and CI discovery. Certain applications are not supported without the use of patterns. See Applications supported by Discovery and Service Mapping for a full list.</td>
<td>In addition to supporting all standard discovery, patterns support:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cloud discovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• configuration file tracking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CI deletion strategies</td>
</tr>
<tr>
<td>The ECC queue and troubleshooting</td>
<td>Discovery creates Multiple ECC Queue records depending on how many probes are launched.</td>
<td>Discovery creates only one additional ECC queue record for the pattern. To troubleshoot any issues with horizontal discovery during identification and exploration, you only have to analyze one ECC queue record for the pattern.</td>
</tr>
<tr>
<td></td>
<td>To troubleshoot any issues with horizontal discovery, you must analyze several ECC queue records to see the data that Discovery retrieved.</td>
<td></td>
</tr>
<tr>
<td>Performance</td>
<td>Depending on what type of target CI you are trying to identify and explore, Discovery might trigger multiple probes during the identification phase, and then trigger additional probes during the discovery phase. It is possible that horizontal discovery with probes can take longer because of the extra overhead in triggering and processing many probes.</td>
<td>Discovery triggers one pattern during the identification an exploration phases. All operations that Discovery needs to run are in the pattern and Discovery can execute them in order until the pattern is finished. Therefore, pattern discovery can lead to faster performance.</td>
</tr>
</tbody>
</table>

Patterns for top-down and for horizontal discovery

Both Discovery and Service Mapping can use the same pattern for horizontal and top-down discovery. But they are edited differently. See Create or modify patterns for all steps. If you take a pattern that was exclusively used for top-down discovery and you want to use it for horizontal discovery, you have to make a few modifications. See Use a pattern for horizontal discovery for instructions.
Add the Horizontal Pattern probe to a classifier

To use a pattern for the identification and exploration phases of horizontal discovery, you must add the Horizontal Pattern probe to the classifiers for the CIs you are trying to discover.

Role required: discovery_admin

**Warning:** This procedure is only intended for custom classifier records or those that do not include a pattern by default. However, if you have already populated your CMDB with data using probes from this classifier and then switch to use patterns, Discovery may create duplicate CIs. Therefore, only follow this procedure if you have never run Discovery with probes for this classifier type.

1. Navigate to Discovery Definition > CI Classification > {classifier type}.
2. Open the classifier record.
3. Click the Triggers probe related list.
4. Deactivate the existing identification and exploration probes.
5. Click Edit, and add the Horizontal Pattern probe. The probe appears in the related list.
6. From the related list view, double click the field under the Pattern column and add the pattern you want to associate with the classification.
7. Remove or deactivate the other probes from the Triggers probe related list.

**Note:** If you delete a pattern, the Horizontal Pattern probe is not automatically removed from the classifier. You must select another pattern for the Horizontal Probe, or you can switch back to using identification and exploration probes specific to the classifier. If you use the Horizontal Probe without a pattern specified, discovery stops after the classification stage.

Use a pattern for horizontal discovery

If you want to use a new pattern, or if you already have a pattern that you were using for top-down discovery, you can use the pattern for horizontal discovery with a few modifications to the relevant classifier.

Role required: discovery_admin

**Warning:** This procedure is only intended for custom classifier records or those that do not include a pattern by default. However, if you have already populated your CMDB with data using probes from this classifier and then switch to use patterns, Discovery may create...
duplicate CIs. Therefore, only follow this procedure if you have never run Discovery with probes for this classifier type.

1. Verify that Discovery can use the pattern:
   a) Navigate to **Pattern Designer > Discovery Patterns**.
   b) Open the pattern. Application patterns that Service Mapping uses are indicated as type **1-Application**.
   c) In the **Identification Sections** on the **Basic** tab, verify that there is at least one section that allows for an entry point type of **TCP** or **All**. If not, create one. See the Identification steps for creating a new pattern.
   d) Save the pattern.

2. On the instance, create or modify the classification for the CI type you want to discover. Configure the classifier as follows:
   a) Navigate to **Discovery Definition > CI Classification > {classification type}**.
   b) Open the relevant classifier.
   c) Configure the classifier as follows:
      * **Relationship type**: Select **Runs on::Runs** (for process classifiers only)
      * **Condition**: Configure the same condition you defined in the pattern.
      * **Triggers probes** Related list: Add the Horizontal Pattern probe, and then add the pattern you are using to the **Pattern** column.

See [Create a Discovery CI classification](#) for a description of the other fields on the classifier.

Run the pattern in **Debug mode** to test it. When you are sure the pattern works, you can run discovery by setting up a discovery schedule or running an on-demand discovery. See [Schedule a horizontal discovery](#) for more information.

### Discovery probes and sensors

Discovery probes and sensors perform data collection and update the CMDB.

**Note:** With each release, patterns are replacing many probes and sensors for Discovery. Consider creating new patterns or editing existing ones if you want to customize what Discovery can find. The information on probes and sensors is intended for customers who are not using patterns yet and for customers who already have customized probes that are retained upon upgrade. See [Patterns and horizontal discovery](#) for more information on patterns.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Discovery phases

Discovery always uses probes and sensors during the first two phases of discovery: **scanning** and **classification**. For the last two phases, **identification** and **exploration**, Discovery can use probes and sensors or patterns. This topic refers to probes and sensors only. See [Discovery basics](#) for
an explanation of these phases. See *Patterns and horizontal discovery* for more information on patterns.

**Probes, sensors, and the ECC queue**

The probe collects the information and the sensor processes it. Both get their instructions from the ECC queue. There is a worker job on the MID Server that monitors the queue for work. The monitor checks for any entries where the Queue is output and the State is ready.

The MID Server then processes all the output ECC messages, runs the necessary probes, and returns the probes results to the ECC queue. These results are put in the ECC Queue as input entries.
ECC queue input

After an entry is inserted in the ECC Queue table, a business rule fires (on insert) that takes that information and runs it through a sensor processor. The sensor processor's job is to take the input data, find any sensors interested in that data, and pass it along to be processed. Those sensors ultimately update the CMDB.
**How probes and sensors work together**

The MID Server launches probes to collect information about a device. The probe sends back information to the sensor to be processed. If the probe has a post-processing script defined, the post-processing script does some data processing on the MID Server before data is sent back to the sensor on the ServiceNow instance. Otherwise, the probe sends back all the data collected and the sensor performs this data processing. In both cases, the sensor updates the CMDB.

A **multi-probe** is a probe that contains probes. A **multi-sensor** processes the data from a multi-probe. To process the data from the multi-probe, the multi-sensor contains individual scripts to process the data returned by each probe contained in the Multiprobe, as well as a main multi-sensor script. The individual scripts pass their processed data to the main multi-sensor script.
### Probe types

<table>
<thead>
<tr>
<th>Device</th>
<th>Probe Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows computers and servers</td>
<td>Remote WMI queries, shell commands</td>
</tr>
<tr>
<td>UNIX and Linux servers</td>
<td>Shell command (via SSH protocol, version 2). Discovery supports any Bourne-compatible shell.</td>
</tr>
<tr>
<td>Storage</td>
<td>CIM/WBEM queries</td>
</tr>
<tr>
<td>Printers</td>
<td>SNMP queries</td>
</tr>
<tr>
<td>Network gear (switches, routers, etc.)</td>
<td>SNMP queries</td>
</tr>
<tr>
<td>Web servers</td>
<td>HTTP header examination</td>
</tr>
<tr>
<td>Uninterruptible Power Supplies (UPS)</td>
<td>SNMP queries</td>
</tr>
</tbody>
</table>

### List of Discovery probes

A wide variety of probes exist for the Discovery application to detect elements on your network. These probes are included with the Discovery application.

The *PPP script* designation in the table indicates whether the probe includes a probe post-processing (PPP) script that runs on the MID Server. The PPP script transforms probe results into a JSON string and returns the string to the ServiceNow instance for sensor processing.

To view probes and their descriptions, navigate to Discovery Definition > Probes.

#### CIM probe

The CIM probe uses WBEM protocols to query a particular CIM server, the CIM Object Manager, for a set of data objects and properties.

For instructions on configuring probe parameters, see Set probe parameters.

The following parameters may be passed to the CIM probe:

#### CIM Probe

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td><em>(Required)</em> The initial host to connect to.</td>
<td>None</td>
</tr>
<tr>
<td>port</td>
<td>The port to connect to. If empty, the value is determined by the &quot;schema&quot; parameter: http = 5988, https = 5989.</td>
<td>5988 or 5989</td>
</tr>
<tr>
<td>schema</td>
<td><em>(Required)</em> The schema to use: &quot;http&quot; or &quot;https&quot;</td>
<td>http</td>
</tr>
<tr>
<td>namespace</td>
<td><em>(Required)</em> The CIM namespace. May be overridden by a query.</td>
<td>None</td>
</tr>
<tr>
<td>queries</td>
<td><em>(Required)</em> A semicolon-delimited list of CIM probe queries to process and return results for.</td>
<td>None</td>
</tr>
<tr>
<td>retries</td>
<td>The number of times to retry a query if it fails due to network connectivity issues.</td>
<td>2</td>
</tr>
</tbody>
</table>
The CIM Intermediate Query Language (CimIQL) uses keys, filters, and dot-walking to traverse the CIM schema.

**Parameter Expansion**

The CIM query language supports standard SNC preprocessed probe parameter expansion. Place variables in queries by encapsulating their names like this:

\[
${\text{foobar}}.\text{CIM\_RunningOS}[0].\text{Name} \\
\text{CIM\_ComputerSystem.}${\text{barfoo}}
\]

The text \${\text{foobar}} is replaced with the contents of the foobars probe parameter passed to the CIM probe; likewise for \text{barfoo}.

**CimIQL**

The CIM Intermediate Query Language (CimIQL) is an intermediate language designed to simplify the process of querying CIM providers.

CimIQL currently supports the standard Web-Based Enterprise Management (WBEM) protocol stack, but others, such as Web Services-Management (WS-MAN), may be added in the future. The query language syntax borrows from elements of Microsoft’s WMI query language and UNIX’s wbemcli command. The CimIQL library is a pure Java implementation.

**Note:** CimIQL is pronounced “simicle”.

**CimIQL Syntax**

CimIQL syntax consists of several elements, including a query and different tokens.

**CimIQL Syntax Element Descriptions**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statement</td>
<td>The most basic element of CimIQL is a valid ,. A statement contains multiple queries delimited by a period . character.</td>
</tr>
<tr>
<td>Query</td>
<td>A \text{represents a single high-level protocol-independent request. Each query is comprised of nested language components and sub-components known as tokens.}</td>
</tr>
<tr>
<td>Token</td>
<td>A \text{describes a specific lexical aspect of the CimIQL syntax.}</td>
</tr>
<tr>
<td>\text{Operation Token}</td>
<td>\text{The first token of each query must be an \text{operation token, which represents the overall logical operation to be performed.}}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection_timeout</td>
<td>The number of milliseconds the probe has to connect to a server.</td>
<td>5000</td>
</tr>
<tr>
<td>socket_timeout</td>
<td>The number of milliseconds the probe has to read data.</td>
<td>5000</td>
</tr>
<tr>
<td>Element</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Component Token</td>
<td>A is a sub-component of an operation token.</td>
<td></td>
</tr>
<tr>
<td>Result</td>
<td>Each query is paired with a result, which is then provided as input to the next query in the statement. A result is comprised of a set of objects and their properties.</td>
<td></td>
</tr>
</tbody>
</table>

CimIQL operation tokens

The CimIQL probe requires operation tokens.

Each of the following core operations has a counterpart in the [CIM Operations over HTTP](https://www.dmtf.org/standards/cim) standard.

**CimIQL Operation Token Summary**

<table>
<thead>
<tr>
<th>Return Value</th>
<th>Details</th>
<th>Equivalent CIM Operation over HTTP</th>
</tr>
</thead>
<tbody>
<tr>
<td>class object</td>
<td>Get Object</td>
<td>GetInstance</td>
</tr>
<tr>
<td></td>
<td>- Retrieves a single object of a specific class by specifying all of its unique keys (as key tokens) and any optional parameter tokens, separated by commas.</td>
<td></td>
</tr>
<tr>
<td>class object</td>
<td>Enumerate Objects</td>
<td>EnumerateInstances</td>
</tr>
<tr>
<td></td>
<td>- Retrieves objects that match a set of condition tokens and parameter tokens.</td>
<td></td>
</tr>
<tr>
<td>class object</td>
<td>Enumerate Associated Objects</td>
<td>Associates</td>
</tr>
<tr>
<td></td>
<td>- Retrieves objects associated with each result from the preceding query.</td>
<td></td>
</tr>
<tr>
<td>statement results</td>
<td>Substitution</td>
<td>Reference the results of a named statement</td>
</tr>
<tr>
<td></td>
<td>- A no-op token that feeds the results of a previous named statement as input into the next query of its own statement.</td>
<td></td>
</tr>
</tbody>
</table>

**Get Object Token**

```
<classname>{<key token>,<parameter token>,...}
```

- Retrieves a single object of a specific class by specifying all of its unique keys (as key tokens) and any optional parameter tokens, separated by commas. This token is also known as the token.
• The `<classname>` is the case-sensitive CIM class name of the desired object. By default, objects of the specified class and of any extended classes are retrieved.
• The key and parameter tokens are enclosed by a single pair of curly brackets `{ ... }.
• This token must only be used as the first query in a statement.
• Returns: class object
• Example:

  ```
  CIM_ComputerSystem{'CreationClassName='Linux_ComputerSystem',Name='runtime'}}.*
  ```

**Enumerate Objects Token**

```classname>[(<condition token>,<parameter token>...,])<array index token> OR
```classname><array index token>```

• Retrieves objects that match a set of condition tokens and parameter tokens. This token is also known as the .
• The condition tokens and parameter tokens are enclosed by two pairs of curly brackets { { ... }}. The curly brackets are optional if there are no conditions or parameters necessary.
• The `<classname>` is the case-sensitive CIM class name of the desired objects. By default, objects of the specified class and of any extended classes are retrieved.
• The index token is optional.
• This token must only be used as the first query in a statement.
• Returns: class object
• Example:

  ```
  CIM_ComputerSystem{['Name='runtime']}}.*
  ```

**Enumerate Associated Objects Token**

```association classname>[(<property filter token>,<parameter token>...,)]<array index token> OR
```association classname><array index token>```

• Retrieves objects associated with each result from the preceding query.
• The condition tokens and parameter tokens are enclosed by two pairs of curly brackets { { ... }}. The curly brackets are optional if there are no properties filters or parameters necessary.
• The `<association classname>` is the name of the many-to-many or one-to-many class that associates two objects together. By default, objects of the specified class and of any extended classes are retrieved.
• The `<parameter token>`, ResultClass, may be specified to filter results based on the resulting object's classname.
• The index token is optional.
• This token must not be used as the first query in a statement.
• Returns: class object
• Example:

  ```
  CIM_ComputerSystem{['Name='runtime']}[2].*
  ```
Substitution Token

${<statement name>}

- A no-op token that feeds the results of a previous named statement as input into the next query of its own statement.
- Returns: void
- Example:

```
$(lastComputer).ElementName
```

CimIQL component tokens

The CimIQL probe requires component tokens, which are sub-components of operational tokens. The following tokens are sub-components of operation tokens.

CimIQL Component Token Summary

<table>
<thead>
<tr>
<th>Token</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Properties token</td>
<td>* OR &lt;property name&gt;,&lt;property name&gt;,...</td>
</tr>
<tr>
<td></td>
<td>- Specifies which properties are to be returned for each object of the final result set.</td>
</tr>
<tr>
<td>Query Delimiter Token</td>
<td>. (Period)</td>
</tr>
<tr>
<td></td>
<td>- Separates queries.</td>
</tr>
<tr>
<td>Index Token</td>
<td>[index]</td>
</tr>
<tr>
<td></td>
<td>- Reduces a preceding query's results to a single object at the specified integer index.</td>
</tr>
<tr>
<td>Key Token</td>
<td>&lt;key name&gt;='value'</td>
</tr>
<tr>
<td></td>
<td>- Matches an object property designated as a key by exact value.</td>
</tr>
<tr>
<td>Condition Token</td>
<td>&lt;property name&gt;&lt;conditional operator&gt;&lt;enclosed value&gt;</td>
</tr>
<tr>
<td></td>
<td>- Matches a single property of an object based on the condition specified.</td>
</tr>
<tr>
<td>Parameter Token</td>
<td>&lt;parameter name&gt;='value'</td>
</tr>
<tr>
<td></td>
<td>- Passes a parameter by &lt;parameter name&gt; to the operation being called. The parameter may be consumed during CimIQL preprocessing or by the CIMOM via request, depending on the parameter.</td>
</tr>
</tbody>
</table>

Properties Token

* OR <property name>,<property name>,...
• Specifies which properties are to be returned for each object of the final result set.
• The wildcard * returns all properties available. Otherwise, each property name desired is provided within a comma-separated list.
• This token is required at the end of each statement.
• Example:

```
CIM_ComputerSystem[0].*
```

**Query Delimiter Token**

, (Period)
• Separates queries.
• Example:

```
CIM_ComputerSystem.PrimaryOwnerContact
```

**Index Token**

(index)
• Reduces a preceding query's results to a single object at the specified integer index.
• This token is always optional.
• Example:

```
CIM_ComputerSystem[0].*
```

**Key Token**

<key name>=<value>
• Matches an object property designated as a key by exact value.
• The <key name> is the name of the property used as a key.
• Example:

```
CIM_ComputerSystem{CreationClassName='Linux_ComputerSystem',Name='runtime'}.*
```

**Condition Token**

<property name><conditional operator><enclosed value>
• Matches a single property of an object based on the condition specified.
• The <property name> is the name of the property to match against.
• The <conditional operator> determines how the property's actual value is compared to its expected value. The operators available are equality (=) and inequality (!=).
• The <enclosed value> should be one of the following:
  • Literal value enclosed in single-quotes ‘...’. For example, foo='bar'
- Regular expression, enclosed by a pair of slashes \(/ \ldots \)/. For example, foo=/bar/*/.

- Example:

```cimql
CIM_ComputerSystem{{Name!='runtime'}}.*
```

### Parameter Token

<parameter name>:©<value>©

- Passes a parameter by `<parameter name>` to the operation being called. The parameter may be consumed during CimIQL pre-processing or by the Common Information Model Object Manager (CIMOM) via request, depending on the parameter.

- Example:

```cimql
CIM_ComputerSystem.CIM_RunningOS{{ResultClass:'Win32_ComputerSystem'}}.*
```

### CimIQL tutorial

This is a tutorial by example where each example builds on the previous example.

<table>
<thead>
<tr>
<th>Order</th>
<th>CimIQL Statement</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>CIM_ComputerSystem(0).*</code></td>
<td>Retrieves the <strong>first result</strong> of all instances of <code>CIM_ComputerSystem</code> and its descendants. Retrieves all <strong>properties</strong>.</td>
</tr>
<tr>
<td>2</td>
<td><code>CIM_ComputerSystem.PrimaryOwnerContact</code></td>
<td>Retrieves all instances of <code>CIM_ComputerSystem</code> and their descendants. Retrieves only one <strong>property</strong>, <code>PrimaryOwnerContact</code>.</td>
</tr>
<tr>
<td>3</td>
<td><code>CIM_ComputerSystem{CreationClassName=©Linux_ComputerSystem©,Name=©runtime©}.*</code></td>
<td>Retrieves a single <strong>unique instance</strong> of <code>CIM_ComputerSystem</code> and its descendants. All <strong>key tokens</strong> must be specified within the <code>{ }</code> <strong>identity token</strong>.</td>
</tr>
<tr>
<td>4</td>
<td><code>CIM_ComputerSystem{{Name!=©runtime©}}.*</code></td>
<td>Retrieves all instances and descendants of <code>CIM_ComputerSystem</code> that do not have a Name <strong>property</strong> of <code>runtime</code>. The <strong>filter token</strong> <code>{ }</code> filters out instances that do not contain all of the properties/keys specified.</td>
</tr>
<tr>
<td>Order</td>
<td>CimIQL Statement</td>
<td>Result</td>
</tr>
<tr>
<td>-------</td>
<td>------------------</td>
<td>--------</td>
</tr>
<tr>
<td>5</td>
<td>CIM_ComputerSystem{{Name=/run.<em>$/}}.</em></td>
<td>Retrieves all instances and descendants of CIM_ComputerSystem that have a value matching the regular expression contained within the / / characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> The regular expression does not require single quotations.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The filter token {{ }} filters out instances that do not contain all of the properties/keys specified.</td>
</tr>
<tr>
<td>6</td>
<td>CIM_ComputerSystem{{Name=runtime}}(2).*</td>
<td>Retrieves the second result of all instances of CIM_ComputerSystem and its descendants where the instances have a property Name of 'runtime'.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The order of operations follows the query syntax.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. <strong>Query</strong> server for all CIM_ComputerSystem and descendants.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. <strong>Filter</strong> results based on Name property.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. <strong>Retrieve the second instance</strong> that passed the filter.</td>
</tr>
<tr>
<td>7</td>
<td>CIM_ComputerSystem.CIM_RunningOS(0).Name</td>
<td>Retrieves the Name property for the first CIM_OperatingSystem instance of each CIM_ComputerSystem instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The middle-token, CIM_RunningOS, is the name of the <strong>Associator class</strong>, not the end-result.</td>
</tr>
<tr>
<td>8</td>
<td>CIM_ComputerSystem.CIM_RunningOS(CentOS/)(0).Name</td>
<td>Retrieves the Name property for the first CIM_OperatingSystem instance of each CIM_ComputerSystem instance, where each CIM_OperatingSystem instance has a Name property containing 'CentOS'.</td>
</tr>
</tbody>
</table>

**CimIQL results**

CIM Probe results are passed to the probe sensor as an XML document embedded within the <output> element.
The following is a commented example of a CimQuery batch result.

```xml
<!-- document root -->
<cimqueryset>
  <!-- A single query and query result. Multiple <cimquery> tags may be provided. -->
  <cimquery>
    <!-- The original query, enclosed by CDATA. -->
    <query><![CDATA[<CIM_ComputerSystem[0].PrimaryOwnerContact]]></query>
    <!-- The resulting data is enclosed within a single <result> tag. -->
    <result>
      <!-- A single class instance result. Multiple <instance> tags may be provided. -->
      <instance>
        <!-- The instance's CIM classname -->
        <_classname>Linux_ComputerSystem</_classname>
        <!-- A set of this instances identifying keys. Always provided, regardless of property filters. -->
        <_key>
          <!-- The instance's CreationClassName -->
          <CreationClassName><![CDATA[Linux_ComputerSystem]]></CreationClassName>
          <!-- The instance's Name -->
          <Name><![CDATA[runtime]]></Name>
        </_key>
        <!-- Each property that matches the query's property filter will be provided here, in the same format as keys; As <PropertyName>VALUE<PropertyName> where VALUE is enclosed as CDATA -->
        <PrimaryOwnerContact><![CDATA[root@runtime]]></PrimaryOwnerContact>
      </instance>
    </result>
  </cimquery>
</cimqueryset>
```

**DNS probe**

DNS probes determine the DNS names for configuration items (CI).

The default DNS Name Resolver probe is a lightweight and fast probe. It achieves its speed by executing in parallel on the MID Server. The probe utilizes DNS lookup by the MID Server host OS to make use of the maintained DNS cache. The probe also leverages the hosts file on the OS.

These performance improvements can be disabled by reverting to the legacy DNS probe with either MID Server properties or parameters. The names of the MID Server properties/parameters are [mid.probe.use_legacy_forward_dns] and [mid.probe.use_legacy_reverse_dns]. For more information, see MID Server parameters and MID Server properties.

**DNS probe definitions**

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS</td>
<td>Determines DNS names for devices. It has the parameter ip_addresses.</td>
</tr>
<tr>
<td>DNSLookupProbe</td>
<td>Resolves the IP address of a host.</td>
</tr>
<tr>
<td>DNSNameResolver</td>
<td>Resolves the DNS name of an IP address.</td>
</tr>
</tbody>
</table>
**Horizontal Pattern probe**

Discovery uses the Horizontal Pattern probe to launch patterns for horizontal discovery.

The Horizontal Pattern probe works with the Horizontal Discovery sensor to enable Discovery to use patterns for discovery. When you see messages in the ECC Queue from this probe, they appear with the ECC queue name **Pattern Launcher**, followed by the name of the pattern. The probe contains a sensor named **Horizontal Discovery Sensor**, which performs the actual updates of the CMDB based on identification rules.

If you create your own device or process classifier and you want to use patterns for discovery, you must specify this probe in the classifier record. You do not need to modify this probe or the Horizontal Discovery sensor.

**Splitting payload**

When Discovery uses patterns to find certain devices like load balancers, large amounts of data could be sent to the ECC Queue. To better handle large amounts of data, the horizontal pattern probe can split the payload into chunks, and then create multiple ECC Queue records.

Control how the MID Server splits payloads and handles payloads using properties. See **MID Server properties** for more information.

**PowerShell probe**

The PowerShell Probe executes PowerShell V2 scripts on the MID Server host.

PowerShell scripts are defined as probe parameters with the filename as the parameter name. It is available as a Probe probe type by specifying PowerShell as the probe's ECC queue topic.

**PowerShell probe parameters**

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>(Required) The initial host to connect to.</td>
<td>none</td>
</tr>
<tr>
<td>&lt;script name&gt;.ps1</td>
<td>(Required) The filename of the PowerShell script to run. Replace &lt;script name&gt; with a valid filename prefix.</td>
<td>none</td>
</tr>
<tr>
<td>powershell_command_parameter_passing</td>
<td>Specifies whether to pass script parameters on the command line. Regardless of this parameter's value, ServiceNow makes all script parameters on the command line automatically available to PowerShell scripts as environment variables.</td>
<td>false</td>
</tr>
<tr>
<td>Parameter name</td>
<td>Description</td>
<td>Default</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>powershell_param_</td>
<td>Passes additional parameters to the PowerShell script to be executed. Each parameter will appear to the script as an environment variable in the format $env:SNC_&lt;script parameter name&gt;. Parameters with this prefix are not considered encrypted and are passed through to the script untouched. Make sure you select the appropriate parameter between powershell_param_ and powershell_. Using the wrong prefix results in errors in the PowerShell execution, which is passed back to the instance in the ECC queue input.</td>
<td>none</td>
</tr>
<tr>
<td>script parameter name</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>powershell_&lt;script</td>
<td>Passes additional parameters to the PowerShell script to be executed. Each parameter will appear to the script as an environment variable in the format $env:SNC_&lt;script parameter name&gt;. The MID Server assumes that any parameter with this prefix is encrypted and attempts to decrypt it. Make sure you select the appropriate parameter between powershell_param_ and powershell_. Using the wrong prefix results in errors in the PowerShell execution, which is passed back to the instance in the ECC queue input</td>
<td>none</td>
</tr>
<tr>
<td>parameter name&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>debug</td>
<td>Enables debug log output during the probe.</td>
<td>false</td>
</tr>
<tr>
<td>credentials_debug</td>
<td>Displays a &lt;credentials_debug&gt; section in the ECC queue, which can help you troubleshoot credentials. If you set this property to true, credential troubleshooting information is output to the ECC queue, even if the credentials succeed.</td>
<td>false</td>
</tr>
</tbody>
</table>
Scripting requirements

Any custom PowerShell scripts must use environment variables to pass any non-Boolean command line parameter. Replace non-Boolean parameters in the Param() portion of the script with script variables of the same name. Define the script variable as part of the environment with an SNC_ prefix. So a string parameter such as this:

```powershell
Param([string]$paramName)
```

Becomes a script variable such as the following:

```powershell
if(test-path env:\SNC_paramName) {
    $paramName = $env:SNC_paramName
}
```

For example, this parameter definition from the PSScript.ps1 script contains several string parameters that need to be redefined as script variables:

```powershell
```

Defining the non-Boolean parameters as script variables would result in this type of script:

```powershell
Param([boolean]$useCred, [boolean]$isDiscovery, [boolean]$debug)
# Copy the environment variables to the params
if(test-path env:\SNC_computer) {
    $computer=$env:SNC_computer
}
if(test-path env:\SNC_script) {
    $script=$env:SNC_script
}
if(test-path env:\SNC_user) {
    $user=$env:SNC_user
    $password=$env:SNC_password
}
```

Create a custom PowerShell probe

You can create your own PowerShell probe and configure probe parameters.

1. Navigate to Discovery Definition > Probes.
2. Click New.
3. Enter the following values.
   - Name: Any unique name
   - Probe Type: Probe
   - ecc_queue_topic: Powershell
4. Right-click the form header and click Save.
5. From the Probe parameters related list, click New.
6. Create a probe parameter record for each parameter.
7. Click Submit.

SCP Relay probe

The SCP Relay Probe copies a single file or the contents of a directory from one host to another, using the MID Server as a relay.
The SCP Relay probe uses the same parameters as SSHCommand. The commands may be sent in or out of the context of a terminal (tty), and with or without sudo (for those commands, such as lsof, that require being executed in the context of root to cough up the information we need). When commands are sent in the context of a terminal, the path is automatically widened to include a set of default paths (and this can be further widened with the path\_override parameter). If the target machine is the local machine, SSH is not used; instead, a local shell is run to execute the command.

For instructions on configuring probe parameters, see [Set probe parameters](#).

The following parameters may be passed to the SCP Relay probe:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>debug</td>
<td>Enables debug logging.</td>
<td>false</td>
</tr>
<tr>
<td>debug_ssh</td>
<td>Enables J2SSH debug logging (into ssh.log, which can get very large - be careful!)</td>
<td>false</td>
</tr>
<tr>
<td>timeout</td>
<td>Sets the socket connection timeout, in ms.</td>
<td>60000</td>
</tr>
<tr>
<td>path_override</td>
<td>Overrides the default paths set before executing a command.</td>
<td>none</td>
</tr>
<tr>
<td>keyboard_interactive</td>
<td>If true, forces the use of keyboard_interactive SSH login mode.</td>
<td>false</td>
</tr>
<tr>
<td>must_sudo</td>
<td>If true, forces the command to be executed through sudo.</td>
<td>false</td>
</tr>
<tr>
<td>run_in_terminal</td>
<td>If false, disables SSH commands from running in an SSH terminal (this will break many commands).</td>
<td>true</td>
</tr>
<tr>
<td>long_runner</td>
<td>If present, indicates a long-running SSH command.</td>
<td>false</td>
</tr>
<tr>
<td>set_path</td>
<td>If false, disables setting the path environment variable before running the command.</td>
<td>true</td>
</tr>
<tr>
<td>rm_override</td>
<td>If present, overrides the default rm command (&quot;/sbin/rm -f&quot;) with the provided value.</td>
<td>none</td>
</tr>
<tr>
<td>source</td>
<td>Source host or IP to copy from.</td>
<td>required</td>
</tr>
<tr>
<td>source_port</td>
<td>SSH port on the source (defaults to 22).</td>
<td>required</td>
</tr>
<tr>
<td>from_file</td>
<td>Name of the file to copy from the source.</td>
<td>required</td>
</tr>
<tr>
<td>target</td>
<td>Target host or IP to copy to.</td>
<td>required</td>
</tr>
<tr>
<td>target_port</td>
<td>SSH port on the target (defaults to 22).</td>
<td>required</td>
</tr>
<tr>
<td>to_file</td>
<td>Name of the file to copy to the target.</td>
<td>required</td>
</tr>
</tbody>
</table>
SNMP probes
The SNMP probes use the SNMP protocol to query a particular device for a list of OIDs, which are then traversed and the results passed back to the sensors.

Discovery supports SNMP versions 1, 2c, and 3. Discovery uses version 1 and 2c by default. You must enable support for version 3. MID Servers support all SNMP protocol versions by default. You can set a MID Server to only support specific versions of SNMP.

SNMP patterns
The Network Router pattern replaces the several SNMP-related probes for the discovery of routers and switches. See Pattern updates for the New York release.

SNMP probe parameters
This list of parameters may be passed to the SNMP probes.

For instructions on configuring probe parameters, see Set probe parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>oid_spec_list</td>
<td>A list of OID specifications, one per line. Each specification must be in one of the following two forms:</td>
<td>required</td>
</tr>
<tr>
<td></td>
<td>• walk (OID): Walks the OID and all its children</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• table (OID) {OID Children}: Walks all entries in the table, returning only the given children (for example, “iso.org.dod.internet”)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(OID Children) refers to a comma-delimited list of child nodes within the entries for the given table. For example, “ifEntry.ifIndex.ifEntry.ifDescr.ifEntry.ifType” are OID children of the table “iso.org.dod.internet.mgmt.mib-2.interfaces.ifTable”. As a convenience, the table entry prefix may be left off. (The preceding children could be specified as “ifIndex.ifDescr.ifType”.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Any child may include a filter qualifier in parentheses. For example, the child “entPhysicalContainedIn(=0)” specifies returning table entries only if the value of “entPhysicalContainedIn” equals 0. The operators supported in the expression are:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>= equals</td>
<td></td>
</tr>
<tr>
<td></td>
<td>!= does not equal</td>
<td></td>
</tr>
<tr>
<td></td>
<td># contains</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If more than one child has a filter expression, a match on any one of the children causes that entry to be read.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Any content including and after a ‘/’ is ignored (comments). Any OID that does not start with “1.3.6.1” or “iso.org.dod.internet” automatically prefixes with “1.3.6.1.” as a convenience.</td>
<td></td>
</tr>
<tr>
<td>source</td>
<td>The IP address or host name of the device to query SNMP on.</td>
<td>required</td>
</tr>
<tr>
<td>index</td>
<td>The index to apply after the community string, for Cisco-style community string indexing (for VLAN interrogation).</td>
<td>0</td>
</tr>
<tr>
<td>credential_id</td>
<td>The sys_id of a specific credential that is preferred for use above the rest. This parameter is for internal use only and is not supported.</td>
<td>none</td>
</tr>
<tr>
<td>credential_tag</td>
<td>The credential tag that must be used. This parameter is for internal use only and is not supported.</td>
<td>none</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
<td>Default Value</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>timeout</td>
<td>The timeout value (in milliseconds) to wait for a response, instead of the default. You can use this parameter to override the <code>mid.snmp.request.timeout</code> SNMP MID Server configuration parameter.</td>
<td>1500</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> When <code>use_getbulk</code> is set to <code>true</code>, the timeout value is for an individual GETBULK request.</td>
<td></td>
</tr>
<tr>
<td>established_session_timeout</td>
<td>The interval (in milliseconds) to wait for a response after at least one response has been received. Longer values can be useful for collecting complete and accurate data. You can use this parameter to override the <code>mid.snmp.session.timeout</code> SNMP MID Server configuration parameter.</td>
<td>500</td>
</tr>
<tr>
<td>debug</td>
<td>Enables debug logging. Set to true for debug mode.</td>
<td>false</td>
</tr>
<tr>
<td>request_interval</td>
<td>The interval (in milliseconds) between successive requests for an OID when a response has not been received, until the timeout (or established_session_timeout) value is reached. If this value is set to at least as long as the timeout (or established_session_timeout) value, then only a single request is sent for any particular OID. If you change the timeout (or established_session_timeout) value, adjust the request_interval at the same time to avoid sending too many or too few requests for the same OID, as appropriate for a given environment.</td>
<td>400</td>
</tr>
<tr>
<td>request_delay</td>
<td>The interval (in milliseconds) between the receipt of a response and the transmission of the next request. The default is 0 (no delay). This value may be set to slow the overall rate of an SNMP query.</td>
<td>0</td>
</tr>
</tbody>
</table>
| result_format      | Returns JSON formatted payloads for these probes:  
  - SNMP - F5 BIG-IP - System  
  - SNMP - Netscaler - System  
  - SNMP - Routing  

  This parameter returns data in a more compact format to prevent sensor failure or memory problems on a node when the payload becomes large. Do not change this value or delete this parameter.  

  **Caution:** Use of this parameter with any other probes causes the sensor to fail.                                                                 | JSON          |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
</table>
| use_getbulk   | Enables the use of SNMP GETBULK requests to retrieve tabular data from SNMP devices instead of using multiple SNMP GETNEXT requests. For tabular data, GETBULK is more efficient. Regardless of the request type, certain devices may not return any results when they are busy with other tasks. This parameter is used to configure at the probe level. GETBULK can also be set for an individual MID Server or globally for all MID servers. Settings are listed in the order of precedence:  
  - SNMP probe parameter  
  - MID Server configuration parameter  
  - MID Server properties  

The established_session_timeout, request_interval, and request_delay parameters are ignored when use_getbulk is set to true. Instead, the retries parameter is available. The timeout configuration is the same one used by use_getscalar.

By default, the following probes use GETBULK requests (the parameter value is true):  
  - SNMP - Switch - Vlan  
  - SNMP - Switch - BridgePortTable  
  - SNMP - Switch - ForwardingTable  
  - SNMP - Switch - SpanningTreeTable  
  - SNMP - Network - ArpTable  
  - SNMP - Layer 2 Protocol Caches  
  - SNMP - F5 BIG IP - System (only for Service Mapping customers)

**Note:** These probes have a **timeout** value of 5000.

| use_getscalar | Enables the use of simplified retrieval and handling of scalar values from SNMP devices.  

The established_session_timeout, request_interval, and request_delay parameters are ignored when use_getscalar is set to true. Instead, the retries and timeout parameters are available. The timeout configuration is the same one used by use_getbulk.

| retries       | The number of additional attempts Discovery makes to complete an individual GETBULK request (see use_getbulk) or a GETNEXT request when the use_getscalar parameter is set to true. | 2 |

**Load a MIB module**

You can load an additional MIB module by creating a new ecc_agent_mib record and attaching the actual MIB file to the record.

MIBs are only loaded during MID Server startup. When the MID Server starts up, the following management information base (MIB) modules are loaded:

- Modules that are bundled with the MID Server software. These MIBs are defined industry standards.
- Modules that are included in each instance. These MIBs are provided by manufacturers for Discovery to extract specific information from a device.

You can view any errors associated with loading a MIB module in the agent log.

1. Navigate to **MID Server > SNMP MIBs**.
2. Check whether dependencies are met. If your new MIB has dependencies on another MIB, the MIB that fills the dependency must exist before you create your new record. Search the existing MIBs to check that the required MIBs are already loaded. If they are not, use this procedure to also add them to the instance.

3. Click **New** to create a new record. The MID Server MIB File form opens to create a new ecc_agent_mib record.

4. Use the following information to fill out the form:
   - **Name**: The name of the MIB.
   - **Version**: The version of the MIB.
   - **Source**: Use this field to note where the MIB was acquired, such as a URL.
   - **Description**: The description that appears in the ecc_agent_mib table.
   - **Active**: This check box denotes whether the MIB module is enabled or disabled in the instance.

5. Click the Add Attachment icon (_attach) in the upper right to attach the actual MIB file to the new record. The MIB name must begin with an alphabetical character. Remaining characters must be one of the following: alphanumeric, hyphen (-), or underscore (_). The file name must not have an extension. You can reference the existing MIBs for examples. Use the actual name of the MIB for both the MIB record name and the attachment name, but it is not required.

SNMP probe MIB modules
A management information base module (MIB) is a database that is used to manage elements in a network.
Your instance includes object definitions from the most common management information base (MIB) files. Before adding a new object definition, consult the list of MIB modules to ensure that the object definitions are not already available. The instance includes object definitions from the most common management information base (MIB) files. The MID Server MIB File (ecc_agent_mib) table is domain separated. You can create versions of these policies that only a MID Server from the same domain can use. For instructions, see Set up domain separation for MID servers.

The following table contains the MIBs that load automatically. The MIBs that are bundled with the MID Server are not visible in the instance. The MIBs that are in the ecc_agent_mib table are included in the instance by default and can be viewed in the MID Server > SNMP MIBs module.

**Important:** MIB files in a ServiceNow instance do not use a file extension. If the glide.attachment.extensions system property in your instance contains a list of file extensions permitted for import, MIB files do not load. To ensure that MIB files can load, add a temporary extension to the files prior to loading, such as .mib, and then list that extension in the glide.attachment.extensions property. When you load the file, a business rule called MIB filename compliance strips this extension from the file before it inserts the record into the ecc_agent_mib table.

**Management Information Base (MIB) module**

<table>
<thead>
<tr>
<th>Bundled with the MID Server</th>
<th>Included in the instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM-MIB</td>
<td>A10 AX MIB</td>
</tr>
<tr>
<td>BRIDGE-MIB</td>
<td>A10 COMMONS MIB</td>
</tr>
<tr>
<td>DISMAN-EVENT-MIB</td>
<td>CISCO-SMI</td>
</tr>
<tr>
<td>ENTITY-MIB</td>
<td>CISCO-STACK-MIB</td>
</tr>
<tr>
<td>EtherLike-MIB</td>
<td>CISCO-TC</td>
</tr>
<tr>
<td>FRAME-RELAY-DTE-MIB</td>
<td>CISCO-VTP-MIB</td>
</tr>
<tr>
<td>HOST-RESOURCES-MIB</td>
<td>F5-BIGIP-COMMON-MIB</td>
</tr>
<tr>
<td>HOST-RESOURCES-TYPES</td>
<td>F5-BIGIP-LOCAL-MIB</td>
</tr>
<tr>
<td>IANA-ADDRESS-FAMILY-NUMBERS-MIB</td>
<td>F5-BIGIP-SYSTEM-MIB</td>
</tr>
<tr>
<td>IANAIfType-MIB</td>
<td>FOUNDRY-SN-AGENT-MIB</td>
</tr>
<tr>
<td>IP-MIB</td>
<td>FOUNDRY-SN-ROOT-MIB</td>
</tr>
<tr>
<td>IP-FORWARD-MIB</td>
<td>LanMgr-Mib-II-MIB</td>
</tr>
<tr>
<td>IPMROUTE-STD-MIB</td>
<td>MSFT-MIB</td>
</tr>
<tr>
<td>IPV6-MIB</td>
<td>NetWare-Host-Ext-MIB</td>
</tr>
<tr>
<td>ISDN-MIB</td>
<td>NetWare-Server-MIB</td>
</tr>
<tr>
<td>RFC-1212</td>
<td>PowerNet-MIB</td>
</tr>
<tr>
<td>RFC-1215</td>
<td>QMS-MIB</td>
</tr>
<tr>
<td>RFC1155-SMI</td>
<td></td>
</tr>
<tr>
<td>RFC1213-MIB</td>
<td></td>
</tr>
<tr>
<td>RFC1398-MIB</td>
<td></td>
</tr>
<tr>
<td>RFC1406-MIB</td>
<td></td>
</tr>
<tr>
<td>RMON2-MIB</td>
<td></td>
</tr>
<tr>
<td>RSVP-MIB</td>
<td></td>
</tr>
</tbody>
</table>
SSHCommand probe

A probe using the ECC queue topic name SSHCommand executes a shell command on the target host, and returns the resulting output to the sensor.

Discovery supports Bourne Shell (sh) and Bourne-again Shell (bash) commands. Enter shell script commands in the probe’s ECC queue name field. The shell script can use variables and file operations supported by the target UNIX shell.

- The SSH engine is active by default on new instances.
- Customers on upgraded instances can manually enable ServiceNow SSH for a particular probe by setting the use_snc_ssh parameter to true. Alternatively, enable it for all probes on the MID Server by setting the MID Server parameter mid.ssh.use_snc to true.

**Note:** To discover network devices, such as routers and switches, use SNMP credentials, not SSH credentials.

SSHCommand parameters

Several parameters are available for the SSHCommand probe.

For instructions on configuring probe parameters, see [Set probe parameters](#).

Parameters Table

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>allow_unsupported_shells</td>
<td>Allows a probe designer to attempt to run a command on a device that does not have a supported shell. With no shell, the following is true:</td>
</tr>
<tr>
<td></td>
<td>- No path information can be provided.</td>
</tr>
<tr>
<td></td>
<td>- No scripts can function, because there is no ability to handle script parameters.</td>
</tr>
<tr>
<td></td>
<td>- The text that is specified in the ECC queue name field of the probe form is the raw command that is run on the device.</td>
</tr>
<tr>
<td></td>
<td>For example, you may design a probe to display the version of a Cisco switch running NX-OS. The command that the switch understands is show version. To accomplish this, put the show version command in the ECC queue name field, and add the parameter name allow_unsupported_shells with a value of true.</td>
</tr>
<tr>
<td></td>
<td>For example, you may design a probe to display the version of a Cisco switch running NX-OS. The command that the switch understands is show version. To accomplish this, put the show version command in the ECC queue name field, and add the parameter name allow_unsupported_shells with a value of true.</td>
</tr>
<tr>
<td></td>
<td>This parameter is only effective for sncssh. It is not supported with j2ssh. Currently supported shells are sh, bash, ksh, csh, and tcsh.</td>
</tr>
<tr>
<td></td>
<td>- Type: string, true or false</td>
</tr>
<tr>
<td></td>
<td>- Default value: false</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| source    | (Required) Specifies the initial host to connect to.  
  - Type: string (URL)  
  - Default value: None |
| port      | Specifies the target port to connect to.  
  - Type: integer (port)  
  - Default value: 22 |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>debug</td>
<td>Specifies whether to write SSH debug information to the log file. The parameter usage depends on whether the ServiceNow SSH client is enabled. When the ServiceNow SSH client is enabled, this parameter functions as follows:</td>
</tr>
<tr>
<td></td>
<td>• Type: string</td>
</tr>
<tr>
<td></td>
<td>• Default value: false</td>
</tr>
<tr>
<td></td>
<td>The following string values are valid for the ServiceNow SSH client:</td>
</tr>
<tr>
<td></td>
<td>• true: Enables SSH debug information in the log file.</td>
</tr>
<tr>
<td></td>
<td>• false: Disables SSH debug information in the log file.</td>
</tr>
<tr>
<td></td>
<td>• &lt;IP Addresses&gt;: Specifies which IP ranges to enable SSH debug information in the log file. You can enter IP addresses in the following formats:</td>
</tr>
<tr>
<td></td>
<td>• An IP range defined by a slash and the number of bits in the subnetwork. For example, the string 10.10.10.0/24 scans 24 bits of IP addresses from 10.10.10.0 to 10.10.10.254.</td>
</tr>
<tr>
<td></td>
<td>• An IP range defined by a dash. For example, the string 10.10.11.0-10.10.11.165 scans the IP addresses from 10.10.11.0 to 10.10.11.165.</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of specific IP addresses. For example the string 10.10.11.200,10.10.11.235 scans the IP addresses 10.10.11.200 and 10.10.11.235.</td>
</tr>
<tr>
<td></td>
<td>• deferred: Specifies to log SSH debug information in memory unless an error or warning occurs. If an error or warning occurs, the platform publishes the debug information to the log file. This ensures that only the part of the log file pertaining to the error or warning is recorded. If no error or warning is detected, the platform deletes the unused log data from memory when the session closes. Each session stores up to 1000 log messages. If the session exceeds 1000 log messages, the deferred log discards the oldest log message to make room for the newest log message.</td>
</tr>
</tbody>
</table>

When the ServiceNow SSH client is disabled, this parameter enables or disables SSH debug information in the log file:

• Type: true | false
• Default value: false
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| debug_ssh       | Specifies whether the legacy SSH client writes debug information into the `agent/logs/ssh.log` log file. This log file can get very large and should be reviewed frequently.  
The ServiceNow SSH client does not use this parameter.  
- Type: true | false  
- Default value: false                                                                                                                                                                                  |
| timeout         | Sets the socket connection timeout for the legacy SSH client.  
The ServiceNow SSH client does not use this parameter.  
- Type: integer (milliseconds)  
- Default value: 60,000                                                                                                                                                                                   |
| path_override   | Specifies how to change the default paths set before executing a command. Type one or more override paths delimited by a colon (:). The default path is `/usr/sbin: /usr/bin: /bin: /sbin`.  
The ServiceNow SSH client accepts the following prefixes in front of the path_overide value:  
- append: Appends the override path to the end of the host’s path. This is the default behavior.  
- replace: Replaces the host path with the path_overide value.  
- prepend: Appends the override path to the front of the host path.  
- Type: string (a colon-separated list of directories)  
- Default value: None                                                                                                                                     |
| keyboard_interactive | Determines whether to enforce keyboard_interactive SSH login mode.  
- Type: true | false  
- Default value: false                                                                                                                                                                      |
| must_sudo       | Determines whether SSH commands run through `sudo`.  
- Type: true | false  
- Default value: false                                                                                                                                                                            |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>run_in_terminal</td>
<td>Determines whether SSH commands run in an SSH terminal.</td>
</tr>
<tr>
<td></td>
<td>• Type: true</td>
</tr>
<tr>
<td></td>
<td>• Default value:</td>
</tr>
<tr>
<td></td>
<td>• ServiceNow SSH client: false</td>
</tr>
<tr>
<td></td>
<td>• Legacy SSH client: true</td>
</tr>
<tr>
<td>set_path</td>
<td>Determines whether the probe is allowed to alter the session’s PATH variable or not. By default, during session setup, the PATH variable is set to /usr/sbin:/usr/bin:/bin:/sbin.</td>
</tr>
<tr>
<td></td>
<td>• Type: true</td>
</tr>
<tr>
<td></td>
<td>• Default value: true</td>
</tr>
<tr>
<td>rm_override</td>
<td>Overrides the default remove command (/bin/rm -f) with the provided value.</td>
</tr>
<tr>
<td></td>
<td>• Type: string</td>
</tr>
<tr>
<td></td>
<td>• Default value: none</td>
</tr>
<tr>
<td>use_snc_ssh</td>
<td>Enables the ServiceNow SSH client. The ServiceNow SSH client is active by default on new instances. Enabling the ServiceNow SSH client disables the legacy SSH client.</td>
</tr>
<tr>
<td></td>
<td>• Type: true</td>
</tr>
<tr>
<td></td>
<td>• Default value: true</td>
</tr>
<tr>
<td>command_timeout_ms</td>
<td>Number of milliseconds an SSH command is allowed to run before timing out (default is configurable per MID server). The legacy SSH client does not use this parameter.</td>
</tr>
<tr>
<td></td>
<td>• Type: integer</td>
</tr>
<tr>
<td></td>
<td>• Default value: value of the mid.ssh.command_timeout_ms MID Server parameter</td>
</tr>
<tr>
<td>channel_timeout_ms</td>
<td>Specifies the amount of time, in milliseconds, that a MID Server waits for activity during processing of an SSH command. If the MID Server does not detect activity in the specified timeout window, this parameter kills the command.</td>
</tr>
<tr>
<td></td>
<td>• Type: integer (milliseconds)</td>
</tr>
<tr>
<td></td>
<td>• Default value: 300000 (5 minutes)</td>
</tr>
</tbody>
</table>
### Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>preserve_sudo_environment</td>
<td>Specifies whether to use sudo to preserve the environment for SSH. This parameter is only effective if the sudo environment on the host being probed supports the –E switch.</td>
</tr>
<tr>
<td></td>
<td>• Type: true</td>
</tr>
<tr>
<td></td>
<td>• Default value: false</td>
</tr>
<tr>
<td>credentials_debug</td>
<td>Displays a <code>&lt;credentials_debug&gt;</code> section in the ECC queue, which can help you troubleshoot credentials. If you set this property to true, credential troubleshooting information is output to the ECC queue, even if the credentials succeed. See <a href="#">Credentials Troubleshooting</a> for more information.</td>
</tr>
</tbody>
</table>

### SSHCommand path

The SSHCommand probe computes the default path from the following sources:

- MID Server parameter: mid.ssh.path_override
- SSH Command probe parameter: path_override
- User’s default path: Shell configuration file

If you set the MID Server path override parameter, Discovery appends this path to the default path. If you set the probe path parameter, Discovery uses this path instead of the default path. Discovery always appends a user's default execution path to the MID Server and probe parameter paths.

### Default Path

By default, the MID Server searches for SSH commands in the following paths:

- `/usr/sbin`
- `/usr/bin`
- `/bin`
- `/sbin`

### Shell script options

The SSHCommand probe supports the following scripting options in the ECC queue name field.

#### Shell Scripting Options Table

<table>
<thead>
<tr>
<th>Summary</th>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td><code>$(variable)</code></td>
<td>Replaces the token with the value of the variable. For example, <code>$(catalina_home)</code> specifies the installation location of a Tomcat server.</td>
</tr>
<tr>
<td>Include File</td>
<td><code>$(File:file_name.sh)</code></td>
<td>Treats the contents of the specified file as a shell script. For example, <code>$(File:findcat.sh)</code> runs the findcat shell script.</td>
</tr>
</tbody>
</table>
SSH commands for Discovery

These tables display the SSH commands run by Discovery probes on target devices, including parameters.

These SSH commands do not require elevated privileges to run.

### Operating systems

#### AIX commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
<tr>
<td>lsol</td>
<td>-TCP -n -P -F pcnft</td>
</tr>
<tr>
<td>lsattr</td>
<td>-El sys0 -a modelname -F value</td>
</tr>
<tr>
<td>lslpp</td>
<td>-Lc</td>
</tr>
<tr>
<td>/etc/ifconfig</td>
<td>-au; netstat -in</td>
</tr>
<tr>
<td>instfix</td>
<td>-l I grep AIX_ML</td>
</tr>
<tr>
<td>oslevel</td>
<td>-r</td>
</tr>
<tr>
<td>echo `lsattr</td>
<td>-El sys0 -a systemid I cut ...</td>
</tr>
</tbody>
</table>

#### Linux commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>arp</td>
<td>-n</td>
</tr>
<tr>
<td>bash</td>
<td>variable</td>
</tr>
<tr>
<td>cat</td>
<td>/sys/hypervisor/compilation/compiled_by</td>
</tr>
<tr>
<td>cat</td>
<td>/sys/devices/virtual/dmi/id/product_uuid</td>
</tr>
<tr>
<td>dmidecode</td>
<td>I grep UUID</td>
</tr>
<tr>
<td>dmidecode</td>
<td>I cat</td>
</tr>
<tr>
<td>dmidecode; echo dummy &gt;</td>
<td>/dev/null</td>
</tr>
<tr>
<td>grep MemTotal</td>
<td>/proc/meminfo</td>
</tr>
<tr>
<td>ifconfig</td>
<td>-a &amp;&amp; route -n</td>
</tr>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
</tbody>
</table>

#### HP-UX commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>ps</td>
<td>-elix</td>
</tr>
<tr>
<td>echo itick_per_usec/D</td>
<td>I adb -k /stand/vm...</td>
</tr>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
<tr>
<td>model</td>
<td></td>
</tr>
<tr>
<td>uname</td>
<td>-l</td>
</tr>
</tbody>
</table>
### Solaris commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>swlist</td>
<td>`-l product</td>
</tr>
<tr>
<td>echo 'memory_installed_in_machine/D'</td>
<td>`</td>
</tr>
<tr>
<td>lsmod</td>
<td><code>-TCP -n -P -F pcfT</code></td>
</tr>
</tbody>
</table>

### Mac commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>system_profiler</td>
<td>• SPHardwareDataType SPMemoryDataType -xml&lt;br&gt;• SPSerialATADataType SPSASDataType&lt;br&gt;• SPFireWireDataType -xml&lt;br&gt;• SPNetworkDataType -xml;ifconfig</td>
</tr>
<tr>
<td>last</td>
<td>`</td>
</tr>
</tbody>
</table>

### UNIX commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>lsmod</td>
<td><code>-TCP -n -P -F pcfT</code></td>
</tr>
<tr>
<td>ps</td>
<td>`</td>
</tr>
<tr>
<td>sh</td>
<td>`</td>
</tr>
<tr>
<td>cat</td>
<td>`</td>
</tr>
<tr>
<td>variable</td>
<td>`</td>
</tr>
<tr>
<td>unname</td>
<td>`</td>
</tr>
<tr>
<td>df</td>
<td>`</td>
</tr>
</tbody>
</table>
Web servers

Apache commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>uptime;env</td>
<td>LC_ALL=C date +%a %b %e %H:%M %Z %Y</td>
</tr>
</tbody>
</table>

Tomcat commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
<tr>
<td>cat</td>
<td>variable</td>
</tr>
<tr>
<td>find -L ${dirname} -name web.xml -print 2&gt;/dev/null</td>
<td></td>
</tr>
</tbody>
</table>

NGINX commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
<tr>
<td>get-nginxconfig</td>
<td></td>
</tr>
<tr>
<td>get-nginxversion</td>
<td></td>
</tr>
</tbody>
</table>

Databases

Oracle commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
</tbody>
</table>

SQL commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>variable</td>
</tr>
<tr>
<td>variable</td>
<td>-V</td>
</tr>
</tbody>
</table>

HBase commands

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>variable</td>
</tr>
</tbody>
</table>
### Applications

**ESX - OS commands**

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
</tbody>
</table>

**F5 - OS commands**

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
</tbody>
</table>

**HAproxy commands**

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
<tr>
<td>variable</td>
<td>-v</td>
</tr>
</tbody>
</table>

**JBoss commands**

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td>-name web.xml -print</td>
</tr>
<tr>
<td>cat</td>
<td>variable</td>
</tr>
</tbody>
</table>

**KVM commands**

<table>
<thead>
<tr>
<th>Command (ECC queue name)</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>sh</td>
<td>variable</td>
</tr>
<tr>
<td>virsh</td>
<td>Various capabilities</td>
</tr>
</tbody>
</table>

**vCenter probes and probe parameters**

vCenter probes scan virtual machines using VMware's vSphere product suite. Each probe scans for different kinds of data, such as networks, NICs, and tags. The VMware - vCenter probe that discovered all vCenter objects in previous releases is deprecated in the Istanbul release and replaced by multiple probes.

### Upgrade changes

In upgraded systems, the vCenter process classifier and the vmapp port probe are configured to trigger the VMware - vCenter Datacenters probe. This probe then triggers the probes that
discover individual vCenter objects such as hosts, storage, and datastores. The legacy probe, VMware - vCenter, is still in the system but is not triggered to run in the updated instance.

**Available vCenter probes and probe parameters**

vCenter probe parameters allow you to disable the probes for the objects you are not interested in discovering. You can also reduce the size of the payload of a probe by specifying a page size.

**Important:** Before disabling a probe, be aware of any dependencies the probe might have. If the probe you disable triggers another probe, the dependent probe is also disabled, and cannot collect data.

**Relationship of vCenter elements to probe parameters**

These parameters are available for vCenter probes.

**Note:** By default, these parameters are not set in the platform, except where noted.

**vCenter probe parameters**

<table>
<thead>
<tr>
<th>Probe</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMWare - vCenter Datacenters</td>
<td>• datacenter_only (Cloud Management)</td>
</tr>
<tr>
<td></td>
<td>• debug</td>
</tr>
<tr>
<td></td>
<td>• disable_vm_probe</td>
</tr>
<tr>
<td></td>
<td>• disable_network_probe</td>
</tr>
<tr>
<td></td>
<td>• disable_datastore_probe</td>
</tr>
<tr>
<td></td>
<td>• disable_cluster_probe</td>
</tr>
<tr>
<td></td>
<td>• datacenters_only</td>
</tr>
<tr>
<td></td>
<td>• fixed_credential_id (Cloud Management)</td>
</tr>
<tr>
<td></td>
<td>• disable_vm_nic_probe</td>
</tr>
<tr>
<td></td>
<td>• disable_host_probe</td>
</tr>
<tr>
<td></td>
<td>• disable_host_storage_probe</td>
</tr>
<tr>
<td></td>
<td>• vcenter_timeout: default 10000 milliseconds (10 seconds).</td>
</tr>
<tr>
<td>VMWare - vCenter VMs</td>
<td>• disable_vm_nic_probe</td>
</tr>
<tr>
<td></td>
<td>• disable_vm_nics_vnics (set to false)</td>
</tr>
<tr>
<td></td>
<td>• disable_vm_nic_vdisks (set to true for Discovery only and false when Cloud Management is enabled)</td>
</tr>
<tr>
<td></td>
<td>• disable_vm_tags_probe</td>
</tr>
<tr>
<td></td>
<td>• refresh_state (Cloud Management)</td>
</tr>
<tr>
<td></td>
<td>• page_size (100)</td>
</tr>
</tbody>
</table>

**Note:** The settings for the vnics and vdisks parameters preserves the default behavior of vCenter Discovery.

VMWare - vCenter VM Tags* none
<table>
<thead>
<tr>
<th>Probe</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMWare - vCenter VM NICs</td>
<td>none</td>
</tr>
<tr>
<td>VMWare - vCenter Networks</td>
<td>page_size (500)</td>
</tr>
<tr>
<td>VMWare - vCenter Datastores</td>
<td>page_size (500)</td>
</tr>
<tr>
<td>VMWare - vCenter Clusters</td>
<td>· disable_host_probe</td>
</tr>
<tr>
<td></td>
<td>· disable_host_storage_probe</td>
</tr>
<tr>
<td></td>
<td>· page_size (1000)</td>
</tr>
<tr>
<td>VMWare - vCenter ESX Hosts</td>
<td>· disable_host_storage_probe</td>
</tr>
<tr>
<td></td>
<td>· page_size (10)</td>
</tr>
<tr>
<td>VMWare - vCenter ESX Hosts Storage</td>
<td>page_size (175)</td>
</tr>
<tr>
<td>VMWare - vCenter ESX Hosts License</td>
<td>none</td>
</tr>
</tbody>
</table>

*For more information, see [VMware tags](#).*

Consider the following when setting these parameters:

- **Disable**: You disable a probe by setting the parameter in the triggering probe. For example, if you are not interested in discovering storage, set the `disable_host_storage_probe` parameter to `true` in the VMWare - vCenter ESX Hosts probe.
- **Page size**: Page size parameters control the number of CIs to discover with a single probe. Use this parameter to limit payload size by reducing the number of vCenter elements discovered at a time by any probe. The page size expressed in parentheses is the default in the base system.
- **Debug**: Set the debug parameter in the VMWare - vCenter Datacenters probe to allow debugging for all the vCenter probes. Debugging returns the raw vCenter data in each probe payload.

Trigger custom probes with the vCenter Discovery extension

vCenter Discovery extension allows you to collect additional attributes of currently discovered CI types or collect attributes of new CI types by triggering custom probes from an existing sensor.

vCenter sensor records support triggering custom probes that use conditional scripts for gathering data not typically collected by Discovery or for discovering CI types that you create.

Role required: admin

**Caution**: The vCenter extension is intended to be used to create a single custom probe that extends existing Discovery functionality. Unless you are a user who is familiar with advanced scripting, do not attempt to configure your new probe/sensor to launch an additional probe.

1. Navigate to Discovery Definition > Sensors.
2. Select a vCenter sensor record.
   It is important to trigger a new probe at the correct time. For example, if you want to discover additional information about virtual machines in your network, trigger your probe from the VMWare - vCenter VMs sensor. If you trigger your custom probe from the VMWare - vCenter Datacenters sensor, your new probe and the existing VM probe execute in parallel. This might prevent VM records from being created or updated.
3. Select the Triggers probes related list.
4. Click New to create a new probe to be triggered by this sensor or Edit to add an existing probe to the list.
You can also use an existing probe record as a template for your new probe by opening the record and selecting the **Insert and Stay** option from the context menu.

5. Click the information icon to the right of the probe name to open the **Conditional Probes Triggered by Sensor** record.

6. View the script.

The script passes the **parms** and **data** arguments to the function, using values passed to them from the triggering sensor. The script configures the **parms** object, from which the probe parameters are derived. The script returns **true** to trigger the probe or **false** to cancel it.

In this example, the VMWare – vCenter Datacenters sensor **parms** are all probe parameters that are supported by the Datacenters probe. The **data** argument is an object containing up to 4 arrays, named **vm**, **datastore**, **network**, and **cluster**.

---

**Conditional Probes Triggered by Sensor record**

```javascript
function(parms, data) {
  var disableNic = g_probe.getParameter('disable_vm_nic_vnic1'),
      disableDisks = g_probe.getParameter('disable_vm_nic_vdisk1'),
      disableNicProbe = g_probe.getParameter('disable_vm_nic_probe');

  // Support the old probe parameter to disable the VM probe
  if ('' + g_probe.getParameter('disable_vm_probe') == 'true')
    return false;

  // Copy VM specific parameters that need to be available to
  // sensors for triggered probes.
  if (disableNic)
    parms.disable_vm_nic_vnic1 = disableNic;
  if (disableDisks)
    parms.disable_vm_nic_vdisk1 = disableDisks;
  if (disableNicProbe)
    parms.disable_vm_nic_probe = disableNicProbe;

  data.vm_ids = JSON.stringify(data.vm);
  return !!data.vm;
}
```
Windows probes and permissions
Discovery accesses devices and software by executing commands as a specific user on Windows computers.

Most probes require access to Windows classes, properties, and registry entries. Certain probes also require additional access to Windows directories and resources. Security policies vary by organization, so there is no one specific role or right to grant. Ensure that the Windows user has local admin permission for these Windows components.

Administrative shares
Windows administrative shares are hidden server resources that Discovery uses to temporarily store the results of processes run by specific probes. The MID Server script file LaunchProc.psm1 launches the process, writes its output to the administrative share on the machine, and then retrieves the results for the MID Server. Access to administrative shares is restricted to users with administrative rights to the target, such as users, local or on the domain, who belong to the local Administrators group.

With the New York release, all Windows probes using WMI protocol call the LaunchProc.psm1 script file and write to admin$ share folder as default. If another network share is mounted on each Window's target, the folder may be changed by updating the MID Server property mid.powershell.target_base_dir.

Windows Classes
Most Windows probes access Windows classes and properties contained in those classes. Some Windows classes do not specify a file namespace path. ServiceNow uses root\cimv2\<Class> by default if an explicit path is not specified.

<table>
<thead>
<tr>
<th>Probe</th>
<th>Windows Class</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hyper-V - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToNode</td>
<td>Dependent</td>
</tr>
<tr>
<td>Hyper-V - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToNode</td>
<td>Antecedent</td>
</tr>
<tr>
<td>Hyper-V - Cluster</td>
<td>root\MSCluster\MSCluster_Resource</td>
<td>Type</td>
</tr>
<tr>
<td>Hyper-V - Resource Pools</td>
<td>root\virtualization\Msvm_ResourcePool</td>
<td>ResourceSubType</td>
</tr>
<tr>
<td>Hyper-V - Resource Pools</td>
<td>root\virtualization\Msvm_ResourcePool</td>
<td>Name</td>
</tr>
<tr>
<td>Hyper-V - Resource Pools</td>
<td>root\virtualization\Msvm_ResourcePool</td>
<td>Capacity</td>
</tr>
<tr>
<td>Hyper-V - Resource Pools</td>
<td>root\virtualization\Msvm_ResourcePool</td>
<td>AllocationUnits</td>
</tr>
<tr>
<td>Hyper-V - Resource Pools</td>
<td>root\virtualization\Msvm_ResourcePool</td>
<td>InstanceID</td>
</tr>
<tr>
<td>Hyper-V - Resource Pools</td>
<td>root\virtualization\Msvm_MemorySettingData</td>
<td>VirtualQuantity</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Class</td>
<td>Property</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ComputerSystem</td>
<td>ElementName</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ComputerSystem</td>
<td>TimeOfLastStateChange</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_LogicalDisk</td>
<td>NumberOfBlocks</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_LogicalDisk</td>
<td>AllocationUnits</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ComputerSystem</td>
<td>ChassisSerialNumber</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_VirtualSystemSettingData</td>
<td>InstanceID</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_VirtualSystemSettingData</td>
<td>BIOSGUID</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ComputerSystem</td>
<td>Description</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ComputerSystem</td>
<td>Name</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_LogicalDisk</td>
<td>BlockSize</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_VirtualSystemSettingData</td>
<td>BIOSSerialNumber</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_VirtualSystemSettingData</td>
<td>BaseBoardSerialNumber</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_VirtualSystemSettingData</td>
<td>SystemName</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_LogicalDisk</td>
<td>ElementName</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_LogicalDisk</td>
<td>SystemName</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ProcessorSettingData</td>
<td>VirtualQuantity</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ProcessorSettingData</td>
<td>InstanceID</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_ComputerSystem</td>
<td>EnabledState</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_SyntheticEthernetPortSettingData</td>
<td>ElementName</td>
</tr>
<tr>
<td>Hyper-V - Virtual Machines</td>
<td>root\virtualization\Msvm_MemorySettingData</td>
<td>InstanceID</td>
</tr>
<tr>
<td>Hyper-V - Virtual Networks</td>
<td>root\virtualization\Msvm_VirtualSwitch</td>
<td>Name</td>
</tr>
<tr>
<td>Hyper-V - Virtual Networks</td>
<td>root\virtualization\Msvm_VirtualSwitch</td>
<td>ElementName</td>
</tr>
<tr>
<td>Hyper-V - Virtual Networks</td>
<td>root\virtualization\Msvm_ActiveConnection</td>
<td>Antecedent</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Class</td>
<td>Property</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------------------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Hyper-V - Virtual Networks</td>
<td>root\virtualization\Msvr_ActiveConnection</td>
<td>Dependent</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>CreationDate</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>CommandLine</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>Priority</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>ProcessId</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>ParentProcessId</td>
</tr>
<tr>
<td>Windows - Active Processes</td>
<td>Win32_Process</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_NODE</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>Win32_ComputerSystem</td>
<td>Domain</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_TO_NODE</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_TO_NODE</td>
<td>Type</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_TO_NODE</td>
<td>PrivateProperties</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\virtualization\Msvr_ComputerSystem</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_TO_NODE</td>
<td>Dependent</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_TO_NODE</td>
<td>Antecedent</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_RESOURCE</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_RESOURCE</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>Win32_OperatingSystem</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>Win32_ComputerSystem</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCLUSTER\MSCLUSTER_NETWORK</td>
<td>AddressMask</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER_RESOURCE</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCLUSTER\MSCLUSTER_CLUSTER</td>
<td>Type</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCLUSTER\MSCLUSTER_NODE</td>
<td>Characteristics</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Class</td>
<td>Property</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------------------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ResourceGroup</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToQuorumResource</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ResourceGroupToResource</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Network</td>
<td>Address</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Resource</td>
<td>PrivateProperties</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Resource</td>
<td>Status</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Node</td>
<td>Flags</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ResourceGroup</td>
<td>PrivateProperties</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ClusterToQuorumResource</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ResourceGroupToResource</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster\MSCluster_NodeToActiveGroup</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Network</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Cluster</td>
<td>Status</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Resource</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Node</td>
<td>State</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ClusterToNode</td>
<td>Dependent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ResourceType</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster\MSCluster_NodeToNetworkInterface</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster\MSCluster_NodeToActiveGroup</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster\MSCluster_Network</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Cluster</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ClusterToNetworkInterface</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_Node</td>
<td>NodeInstanceID</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster\MSCluster_ClusterToNode</td>
<td>Antecedent</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Class</td>
<td>Property</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
<td>----------</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ResourceType</td>
<td>PrivateProperties</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_NodeToNetworkInterface</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ResourceToPossibleOwner</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Node</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_NetworkInterface</td>
<td>Address</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToNetworkInterface</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Node</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_NodeToActiveResource</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToNetworkInterface</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Node</td>
<td>Status</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_NodeToActiveResource</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Resource</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_NetworkInterface</td>
<td>Network</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToNetwork</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Node</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Resource</td>
<td>DeviceID</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_NetworkInterface</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ClusterToNetwork</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_Node</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ResourceToDependentResource</td>
<td>Dependent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster\MSCluster_ResourceGroup</td>
<td>State</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Class</td>
<td>Property</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------------------------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ClusterToResourceGroup</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ResourceToResource</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_Cluster</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_NetworkInterface</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ClusterToResource</td>
<td>PartComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_Node</td>
<td>PrivateProperties</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ResourceToDependent</td>
<td>AntecedentResource</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ResourceGroup</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ClusterToResourceGroup</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - Cluster</td>
<td>root\MSCluster \MSCluster_ResourceTypeToResource</td>
<td>GroupComponent</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>DeviceLocator</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>TypeDetail</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>FormFactor</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>MemoryType</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>DataWidth</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>TotalWidth</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>BankLabel</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>Status</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_Processor</td>
<td>Manufacturer</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_Processor</td>
<td>NumberOfCores</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>Speed</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_Processor</td>
<td>MaxClockSpeed</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_Processor</td>
<td>NumberOfLogicalProcessor</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>SerialNumber</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_Processor</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>PartNumber</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>Capacity</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>Manufacturer</td>
</tr>
<tr>
<td>Windows - CPU / Memory</td>
<td>Win32_PhysicalMemory</td>
<td>Tag</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>Size</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>FreeSpace</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Class</td>
<td>Property</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>DeviceID</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>FileSystem</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>DriveType</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>Description</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>VolumeSerialNumber</td>
</tr>
<tr>
<td>Windows - Disks</td>
<td>Win32_LogicalDisk</td>
<td>VolumeName</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebServerSetting</td>
<td>SecureBindings</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebServerSetting</td>
<td>Name</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebServerSetting</td>
<td>LogFileDirectory</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebInfo</td>
<td>MinorIISVersionNumber</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebInfo</td>
<td>MajorIISVersionNumber</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebServerSetting</td>
<td>ServerComment</td>
</tr>
<tr>
<td>Windows - Get IIS Information</td>
<td>root \MicrosoftIISv2\IIsWebServerSetting</td>
<td>ServerBindings</td>
</tr>
<tr>
<td>Windows - Hardware Information</td>
<td>Win32_ComputerSystemProduct</td>
<td>UUID</td>
</tr>
<tr>
<td>Windows - Hardware Information</td>
<td>Win32_ComputerSystemProduct</td>
<td>IdentifyingNumber</td>
</tr>
<tr>
<td>Windows - Hardware Information</td>
<td>Win32_SystemEnclosure</td>
<td>ChassisTypes</td>
</tr>
<tr>
<td>Windows - Hardware Information</td>
<td>Win32_BIOS</td>
<td>SerialNumber</td>
</tr>
<tr>
<td>Windows - Hardware Information</td>
<td>Win32_SystemEnclosure</td>
<td>SerialNumber</td>
</tr>
<tr>
<td>Windows - Hardware Information</td>
<td>Win32_BaseBoard</td>
<td>SerialNumber</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>Win32_OperatingSystem</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>Index</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>DHCPEnabled</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>MACAddress</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>IPSubnet</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>IPAddress</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapter</td>
<td>Index</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>Caption</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapter</td>
<td>Manufacturer</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>DefaultIPGateway</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapter</td>
<td>NetConnectionID</td>
</tr>
<tr>
<td>Windows - Network</td>
<td>Win32_NetworkAdapterConfiguration</td>
<td>Enabled</td>
</tr>
<tr>
<td>Windows - OS Information</td>
<td>Win32_OperatingSystem</td>
<td>CSDVersion</td>
</tr>
<tr>
<td>Windows - OS Information</td>
<td>Win32_OperatingSystem</td>
<td>Version</td>
</tr>
<tr>
<td>Windows - OS Information</td>
<td>Win32_ComputerSystem</td>
<td>UserName</td>
</tr>
</tbody>
</table>
### Windows Registry entries

Several Windows Registry entries are available for Discovery Windows probes.

**Note:** The Windows - Installed Software sensor appends a timestamp of 00:00:00 to the install_date retrieved from the registry. The installation time of all Windows software is independent of the timezone and is set to midnight of the day it was installed. For example, an install date of **2.19.2017** in the Windows registry appears as **2.19.2017 00:00:00** in the CMDB.

<table>
<thead>
<tr>
<th>Probe</th>
<th>Windows Registry Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - Classify</td>
<td>HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Services/Tcpip/Parameters/Hostname</td>
</tr>
<tr>
<td>Windows - Classify</td>
<td>HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Services/Tcpip/Parameters/Domain</td>
</tr>
<tr>
<td>Windows - Find APD File Location</td>
<td>HKEY_LOCAL_MACHINE/SOFTWARE/APD/APD/CONFIGPATH</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Registry Entries</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Wow6432Node/Microsoft/Windows/CurrentVersion/Installer/UserData/<em>/Products/</em>/InstallProperties/InstallDate</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows NT/CurrentVersion/ProductId</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Installer/UserData/<em>/Products/</em>/InstallProperties/InstallDate</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/DisplayName</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/UninstallString</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Wow6432Node/Microsoft/Windows/CurrentVersion/Installer/UserData/<em>/Products/</em>/InstallProperties/Publisher</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/Version</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/Publisher</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Wow6432Node/Microsoft/Windows/CurrentVersion/Installer/UserData/<em>/Products/</em>/InstallProperties/ProductName</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/ProductName</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Office/<em>/Registration/</em>/ProductID</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Installer/UserData/<em>/Products/</em>/InstallProperties/ProductID</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/DisplayVersion</td>
</tr>
<tr>
<td>Probe</td>
<td>Windows Registry Entries</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Wow6432Node/Microsoft/Office/<em>/Registration/</em>/DigitalProductID</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Installer/UserData/<em>/Products/</em>/InstallProperties/Publisher</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/DisplayName</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/InstallString</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Office/<em>/Registration/</em>/DigitalProductID</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Internet Explorer/Registration/ProductId</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/Publisher</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Internet Explorer/svcVersion</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Office/<em>/Registration/</em>/ProductID</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/InstallDate</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows/CurrentVersion/Uninstall/*/ParentDisplayName</td>
</tr>
<tr>
<td>Windows - Installed Software</td>
<td>HKEY_LOCAL_MACHINE/Software/Microsoft/Windows NT/CurrentVersion/DigitalProductID</td>
</tr>
</tbody>
</table>

**WMIRunner probe**

WMIRunner is a probe type that fetches data from Windows operating systems via the Windows Management Instrumentation (WMI) interface.

The probe handles multiple user-specified WMI Paths to be queried, using a basic form of native WMI query. Each field to be probed must be uniquely named (within the domain of the probe). The probe results returned to the sensor will provide the data found for each field queried, indexed by its name.
When creating a WMI probe, the probe type must be set to WMI Probe and the ECC Queue Topic must be set to WMIRunner.

For instructions on configuring probe parameters, see Set probe parameters.

The following parameters may be passed to the WMI Probe:

### WMIRunner probe parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>Host to connect to.</td>
<td></td>
</tr>
<tr>
<td>port</td>
<td>Port to connect to.</td>
<td></td>
</tr>
<tr>
<td>debug</td>
<td>Enables debug logging.</td>
<td>false</td>
</tr>
<tr>
<td>wmi_timeout</td>
<td>Timeout for the actual WMI probe, in seconds. Use this parameter to change the timeout interval for individual Windows probes. This value overrides the value in the windows_probe_timeout MID Server parameter, which sets a timeout for all probes launched by a specific MID Server. Windows - Installed Software probe is configured with a timeout value of 15 minutes.</td>
<td>300 (seconds)</td>
</tr>
<tr>
<td>process_timeout</td>
<td>Timeout for the process running the script, in seconds. This parameter is for internal use only and is not supported.</td>
<td>wmi_timeout + 10 (seconds)</td>
</tr>
<tr>
<td>credentials_debug</td>
<td>Displays a &lt;credentials_debug&gt; section in the ECC queue, which can help you troubleshoot credentials. If you set this property to true, credential troubleshooting information is output to the ECC queue, even if the credentials succeed. See Credentials troubleshooting for more information.</td>
<td>false</td>
</tr>
</tbody>
</table>

**Note:** The default timeout for WMI/Powershell is 5 minutes, except for the Windows Installed Software probe, which has a default timeout value of 15 minutes. Adding `wmi_timeout` to a probe parameter can change the default timeout of a Windows probe.

**Port probes**

Port probes are used in Discovery by the Shazzam probe to detect protocol activity on open ports on devices it encounters.

When a port probe encounters a protocol in use, the Shazzam sensor checks the port probe record to determine which classification probe to launch. The common protocols WMI, SSH, SNMP, and HTTP in the base system have priority numbers that control the order in which they are launched.

The priority is as follows:

- 1 - WMI
2 - SSH
3 - SNMP
4 - HTTP

In the base system, the WMI probe is always launched first, and if it is successful on a device, no other port probes are launched for that device. If the WMI probe is not successful, then the SSH probe is launched to gather information on the device. If it is not successful, the SNMP probe is launched. This method allows Discovery to classify a device correctly if the device is running more than one protocol (e.g. SSH, SNMP, and HTTP).

**Discovery Port Probe form**

To access the Port Probe form, navigate to **Discovery Definition > Port Probes**.
The Port Probe form provides the following fields:
## Port probes

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Simple name for the port probe that reflects its function (e.g. snmp).</td>
</tr>
<tr>
<td>Description</td>
<td>Definition of the acronym for the protocol. (e.g. ssh is Secure Shell Login).</td>
</tr>
<tr>
<td>Scanner</td>
<td>Shazzam techniques for exploring a port. Some of these are protocol specific, and others are generic. For example, a WMI port probe will use a Scanner value of Generic TCP, and the snmp port probe uses a value of SNMP.</td>
</tr>
<tr>
<td>Active</td>
<td>Indicates whether this port probe is enabled or disabled.</td>
</tr>
<tr>
<td>CIs</td>
<td>Indicates whether this port probe is enabled or disabled for discovering “Configuration Items”.</td>
</tr>
<tr>
<td>IPs</td>
<td>Indicates whether this port probe is enabled or disabled for discovering “IP addresses”.</td>
</tr>
<tr>
<td>Triggered by services</td>
<td>Indicates which services define the port usage. Use this setting to define non-standard port usage and pair the port number with the protocol.</td>
</tr>
<tr>
<td>Triggers probe</td>
<td>Indicates which probe is triggered by the results of this port probe. This is the name of the appropriate classify probe.</td>
</tr>
<tr>
<td>Use classification</td>
<td>Names the appropriate classification table, based on the protocol being explored.</td>
</tr>
<tr>
<td>Classification priority</td>
<td>Establishes the priority in which this port probe runs. If the first port probe fails, then the next probe runs on the device, and so forth, until the correct data is returned. This allows for the proper classification of a device that has two running protocols, such as SSH and SNMP. The default priorities for the Discovery protocols are:</td>
</tr>
<tr>
<td></td>
<td>· 1 - WMI</td>
</tr>
<tr>
<td></td>
<td>· 2 - SSH</td>
</tr>
<tr>
<td></td>
<td>· 3 - SNMP</td>
</tr>
<tr>
<td></td>
<td>· 4 - HTTP</td>
</tr>
<tr>
<td>Supplementary</td>
<td>Launches supplementary classifications after a higher-priority identification succeeds, once again in order of priority.</td>
</tr>
<tr>
<td>Conditional</td>
<td>Runs this port probe if any one of the non-conditional probes return an open port. The conditional port probes in the out-of-box system attempt to resolve the names of Windows devices and DNS names. These ports probes take additional resources and are not used unless activity is detected on open ports.</td>
</tr>
<tr>
<td>Script</td>
<td>Script to run.</td>
</tr>
</tbody>
</table>

Shazzam probe, port probes, and protocols
Port scanning is the first step in the Discovery process. The Shazzam probe performs port scanning, regardless of whether you use patterns for horizontal discovery. The following table lists the known ports and protocols used by Discovery.

Several port probes are available in the base system. Each port probe uses an IP Service, which is a record that tells Discovery which port to use for a specific protocol. Review this table before you block any ports with a firewall.

**Caution:** Make sure that you do not block any ports that Discovery needs.

### Default port probes and default IP services

<table>
<thead>
<tr>
<th>Default port probe name</th>
<th>Default classification</th>
<th>Default IP Service, protocol and port</th>
</tr>
</thead>
<tbody>
<tr>
<td>dns</td>
<td>Process Classification (discovery_classy_proc)</td>
<td>dns (port 53)</td>
</tr>
<tr>
<td>http</td>
<td>HTTP Classification (discovery_classy_http)</td>
<td>http (port 80) and https (port 443)</td>
</tr>
<tr>
<td>ip_phone</td>
<td>SNMP Classification (discovery_classy_snmp)</td>
<td>sip (port 5060)</td>
</tr>
<tr>
<td>osx</td>
<td>Scan Results Application Classifier (discovery_classy_scan_app)</td>
<td>afp (port 548)</td>
</tr>
<tr>
<td>printer</td>
<td>Scan Results Application Classifier (discovery_classy_scan_app)</td>
<td>hp-pdl-datastr (port 9100) and printer (port 515)</td>
</tr>
<tr>
<td>sip</td>
<td>Process Classification (discovery_classy_proc)</td>
<td>sip (port 427)</td>
</tr>
<tr>
<td>snmp</td>
<td>SNMP Classification (discovery_classy_snmp)</td>
<td>snmp (port 161)</td>
</tr>
<tr>
<td>ssh</td>
<td>UNIX Classification (discovery_classy_unix)</td>
<td>ssh (port 22)</td>
</tr>
<tr>
<td>vmapp</td>
<td>Application Classification (discovery_classy_appl)</td>
<td>vmapp_https (port 5,480) and vmapp6_https (port 9,443)</td>
</tr>
<tr>
<td>wbem</td>
<td>CIM Classification (discovery_classy_cim)</td>
<td>wbem_https (port 5989)</td>
</tr>
<tr>
<td>winrm</td>
<td>Windows Classification (discovery_classy_windows)</td>
<td>winrm (port 5,985) and winrm_ssl (port 5,986)</td>
</tr>
<tr>
<td>wins</td>
<td>Process Classification (discovery_classy_proc)</td>
<td>ms-nb-ns (port 137)</td>
</tr>
<tr>
<td>wmi</td>
<td>Windows Classification (discovery_classy_windows)</td>
<td>epmap (port 135)</td>
</tr>
</tbody>
</table>

This table shows you other common ports and protocols that Discovery uses.

### Discovery ports and protocols

<table>
<thead>
<tr>
<th>Name</th>
<th>Service name</th>
<th>Port</th>
<th>Details</th>
<th>Creates</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>afp</td>
<td>Apple File Protocol</td>
<td>548</td>
<td></td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>Name</td>
<td>Service name</td>
<td>Port</td>
<td>Details</td>
<td>Creates</td>
<td>Protocol</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------</td>
<td>------</td>
<td>----------------------------------------------</td>
<td>--------------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>BEA Weblogic</td>
<td></td>
<td>7001</td>
<td>cmdb_ci_app_server</td>
<td>TCP/UDP</td>
<td></td>
</tr>
<tr>
<td>dns</td>
<td>Domain Name Service</td>
<td>53</td>
<td>To resolve the name of each IP Address</td>
<td></td>
<td>TCP/UDP</td>
</tr>
<tr>
<td>epmap</td>
<td>Microsoft RPC (WMI, DCOM)</td>
<td>135</td>
<td>Windows Systems</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>ftp</td>
<td></td>
<td>21</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>hp-pdl-datastr</td>
<td>Printer PDL Data Stream</td>
<td>9100</td>
<td>HP Printers</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>http</td>
<td>HyperText Transfer Protocol</td>
<td>80</td>
<td>Web Servers</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>https</td>
<td>HyperText Transfer Protocol over Secure Socket</td>
<td>443</td>
<td>Secure Web Servers</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>IBM DB2</td>
<td></td>
<td>50000</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>IBM MQSeries</td>
<td></td>
<td>1414</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>IBM Websphere</td>
<td></td>
<td>9080</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>IBM Websphere SSL</td>
<td></td>
<td>9443</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>IMAPS</td>
<td></td>
<td>993</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>pip (Internet Print Protocol)</td>
<td>IP Phone/Session Initiation Protocol</td>
<td>5060</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>LDAP</td>
<td></td>
<td>389</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>LDAPs</td>
<td></td>
<td>636</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>Microsoft netbios</td>
<td></td>
<td>139</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>Microsoft-ds</td>
<td></td>
<td>445</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>ms-nb-ns</td>
<td></td>
<td>137</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>Microsoft SQL server</td>
<td></td>
<td>1433</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>MySQL</td>
<td></td>
<td>3306</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>Nagios NRPE</td>
<td></td>
<td>5666</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>nfs</td>
<td></td>
<td>2049</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP/UDP</td>
</tr>
<tr>
<td>Oracle TNS</td>
<td></td>
<td>1521</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>POP3</td>
<td></td>
<td>110</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>postgresql</td>
<td></td>
<td>5432</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP/UDP</td>
</tr>
<tr>
<td>printer</td>
<td>Printer</td>
<td>515</td>
<td>cmdb_ci_web_server</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>Name</td>
<td>Service name</td>
<td>Port</td>
<td>Details</td>
<td>Creates</td>
<td>Protocol</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------</td>
<td>------</td>
<td>---------------------------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>sip</td>
<td>SIP (Session Initiation Protocol)</td>
<td>5060</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sip</td>
<td>Service Location Protocol (SLP)</td>
<td>427</td>
<td>TCP/UDP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>smtp</td>
<td>TCP</td>
<td>25</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>smux (SNMP multiplexing)</td>
<td></td>
<td>199</td>
<td>TCP/UDP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>snmp</td>
<td>Simple Network Management Protocol</td>
<td>161</td>
<td>UDP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>snmptrap</td>
<td></td>
<td>162</td>
<td>UDP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ssh</td>
<td>Secure Shell Service</td>
<td>22</td>
<td>Unix Systems</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>sunrpc</td>
<td></td>
<td>111</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>telnet</td>
<td></td>
<td>23</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIBCO Rendezvous</td>
<td></td>
<td>7500</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tomcat HTTP</td>
<td></td>
<td>8080</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vmapp6=https</td>
<td></td>
<td>9443</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vmapp=https</td>
<td>vCenter Server Appliance Web Interface using https</td>
<td>5480</td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>wbem=https</td>
<td>CIM-XML via HTTPS(WBEM)</td>
<td>5989</td>
<td>CIM Classification</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>wins</td>
<td>Windows Internet Name Service</td>
<td>137</td>
<td>NetBIOS Name Resolver</td>
<td></td>
<td>UDP</td>
</tr>
</tbody>
</table>

**Windows and dynamic ports**

Supported Windows machines can have dynamic ports ranges: 49152-65535 for both TCP and UDP.

**Configure Shazzam probe**

When you run Discovery, the Shazzam probe finds your active network devices by scanning specified ports on specified IP address ranges. If the list of IP ranges being scanned is large, you can configure the Shazzam payload for JSON encoding to reduce its size.

**Role required: admin**

You control the behavior of individual Shazzam probes using basic and advanced parameters.

For instructions on configuring probe parameters, see [Set probe parameters](#).
**Note:** The Shazzam payload is controlled by the following **MID Server properties**. These properties do not need to be added to the MID Server. The Shazzam probe will fall back on their default values.

- The `shazzam.chunk_size` property controls the maximum number of IP addresses Shazzam can scan in parallel.
- The `mid.shazzam.regulator.interval_ms` property sets the interval in which Shazzam can launch packets.
- The `mid.shazzam.regulator.packets_per_interval` property sets the number of packets that Shazzam can launch in that time interval.

1. Navigate to **Discovery Definition > Probes**.
2. Select **Shazzam**.
3. Add or edit parameters in the **Probe Parameters** related list.
4. Configure the Shazzam parameters.

These parameters are available for fine-tuning the Shazzam probe. These values are defined in the probe record only.

### Shazzam advanced parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>BannerTCP_waitForConnectMS</code></td>
<td>Sets the number of milliseconds the BannerTCP scanner waits for a connection and banner. Default: 1500</td>
</tr>
<tr>
<td><code>debug</code></td>
<td>Enables debug logging if set to <code>true</code>. Default: <code>false</code></td>
</tr>
<tr>
<td><code>delay_webem</code></td>
<td>Delays classification of systems with a WBEM port open until the final Shazzam sensor job. For a large schedule discovering many WBEM ports, delaying classification until the last sensor job could cause the node to run out of memory. Setting this parameter to <code>false</code> allows classification of these systems to occur across all Shazzam sensor jobs. Default: <code>true</code></td>
</tr>
<tr>
<td><code>DNS_alternativePort</code></td>
<td>Deprecated</td>
</tr>
<tr>
<td><code>DNS_waitForResponseMS</code></td>
<td>Sets the number of milliseconds the DNS scanner waits for a response. Default: 1000</td>
</tr>
<tr>
<td><code>GenericTCP_waitForConnectMS</code></td>
<td>Sets the number of milliseconds the GenericTCP scanner waits for a connection. Default: 1000</td>
</tr>
<tr>
<td><code>HTTP_waitForConnectMS</code></td>
<td>Sets the number of milliseconds the HTTP scanner waits for a connection. Default: 500</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP_waitForResponseMS</td>
<td>Sets the number of milliseconds the HTTP scanner waits for a response.</td>
</tr>
<tr>
<td></td>
<td>Default: 500</td>
</tr>
<tr>
<td>NBT_alternativePort</td>
<td>Deprecated</td>
</tr>
<tr>
<td>NBT_waitForResponseMS</td>
<td>Sets the number of milliseconds the NBT scanner waits for a response.</td>
</tr>
<tr>
<td></td>
<td>Default: 500</td>
</tr>
<tr>
<td>report_inactive</td>
<td>When true, reports devices that are alive but inactive. For example, a device has no ports open but refuses at least one port connection request.</td>
</tr>
<tr>
<td></td>
<td>Default: true</td>
</tr>
<tr>
<td>scanner_log</td>
<td>Enables scanner logging if set to true. This logging information appears in the Shazzam probe response.</td>
</tr>
<tr>
<td></td>
<td>Default: false</td>
</tr>
<tr>
<td>shazzam_report_dead</td>
<td>When true, reports devices with dead IP addresses. For example, a device that has all ports closed.</td>
</tr>
<tr>
<td></td>
<td>Default: false</td>
</tr>
<tr>
<td>SNMP_alternativePort</td>
<td>Deprecated</td>
</tr>
<tr>
<td>SNMP_tapIntervalMS</td>
<td>Sets the number of milliseconds the SNMP scanner waits between taps.</td>
</tr>
<tr>
<td></td>
<td>Default: 1000</td>
</tr>
<tr>
<td>SNMP_taps</td>
<td>Sets the number of taps (requests) the SNMP scanner attempts.</td>
</tr>
<tr>
<td></td>
<td>Default: 2</td>
</tr>
<tr>
<td>SNMP_waitForResponseMS</td>
<td>Sets the number of milliseconds the SNMP scanner waits for a response after the last tap.</td>
</tr>
<tr>
<td></td>
<td>Default: 1000</td>
</tr>
</tbody>
</table>

**Control Shazzam payload size**

A system property converts Shazzam payloads into JSON strings, which dramatically reduces their size. This setting prevents nodes from running out of memory when a single schedule discovers large numbers of IP ranges.

**Role required:** admin

The `glide.discovery.shazzam_ranges_json` property is set to `true` for new instances. This setting encodes the payload as a JSON string. The property is configurable by administrators and is available in the Discovery Definition > Properties module. The property label is Use JSON for IP ranges in Shazzam in the module.

This property is set to `false` in upgraded instances and is not visible by default. Adding the property manually to your upgraded instance enables the feature, but does not add it to the...
Discovery Definition > Properties module. To enable JSON encoding and add it to the module, import the update set attached to KB0687626.

1. In the navigation filter, type sys_properties.list and press Enter.
2. In the list of system properties, click New.
3. Complete the form, using these field values:
   - **Name**: glide.discovery.shazzam_ranges_json
   - **Description**: Encodes the Shazzam payload in JSON.
   - **Value**: true
4. Click Submit.

Discovery sensors

Every probe in Discovery must have a corresponding sensor to process the data returned. For example, if incoming data is the result of a WMI probe, then the WMI sensor is triggered to process the payload.

**Note:** If you create a multiprobe, you must create a multisensor to process the data returned from this probe. For details, see Multiprobes and Multisensors.

Navigate to Discovery > Discovery Definition > Sensors and edit or create a sensor.

**Sensor Fields Table**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique probe name.</td>
</tr>
<tr>
<td>Reacts to probe</td>
<td>Select the probe whose payload this sensor must process.</td>
</tr>
<tr>
<td>Sensor Class (sys_class_name)</td>
<td>Select the type of sensor to create:</td>
</tr>
<tr>
<td></td>
<td>· Import Export Map: This option is not currently used.</td>
</tr>
<tr>
<td></td>
<td>· MultiSensor: Multisensors process the data returned from multiprobes.</td>
</tr>
<tr>
<td></td>
<td>· Select this type to create a multisensor that responds to the simple probes used in a multiprobe.</td>
</tr>
<tr>
<td></td>
<td>· Sensor: Select this type if this sensor is a simple sensor that responds to a simple probe.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter an optional description of the sensors function.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Sensor type (sensor_type)</td>
<td>Specify how the results from the probe are processed.</td>
</tr>
<tr>
<td></td>
<td>• Classifier: This field is not currently used.</td>
</tr>
<tr>
<td></td>
<td>• Java: This field is not currently used.</td>
</tr>
<tr>
<td></td>
<td>• Javascript: Returned data from the probe is processed in the sensor itself, outside the application, and is visible to the user. This is the most common sensor type.</td>
</tr>
<tr>
<td></td>
<td>• Multiline Text: This field is not currently used.</td>
</tr>
<tr>
<td></td>
<td>• XML: The XML data from the probe is broken into pieces. Some pieces can be used to launch other probes that the original sensor needs to complete all the necessary information about a device.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this option to enable or disable the sensor.</td>
</tr>
<tr>
<td>Script</td>
<td>Enter a script to run when processing the probe. You can use the g_probe_parameters hashmap in a sensor script to set probe parameters for any configured, triggered probes. For example, this code sets a <code>node_port</code> parameter to 16001 for all triggered probes.</td>
</tr>
<tr>
<td></td>
<td><code>g_probe_parameters['node_port'] = 16001;</code></td>
</tr>
<tr>
<td>Responds to probes</td>
<td>Use this related list to specify the simple probe within a multiprobe whose payload this multisensor must process. This list is available for multisensors only.</td>
</tr>
<tr>
<td>Triggers probes</td>
<td>Use this related list to specify which probes this sensor can start for additional exploration.</td>
</tr>
<tr>
<td>These fields can be added by configuring the form</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>Agent</td>
<td>Enter the ECC queue type to match this sensor with.</td>
</tr>
<tr>
<td>Apply Defaults</td>
<td>Select this option to apply default sensor values.</td>
</tr>
<tr>
<td>Classification type</td>
<td>Select the classification type associated with this sensor.</td>
</tr>
<tr>
<td>Coalesce</td>
<td>Enter the field to match against when determining whether to coalesce.</td>
</tr>
<tr>
<td>Condition script</td>
<td>Create a predetermined requirement for this sensor to run.</td>
</tr>
<tr>
<td>Data option</td>
<td>Select how to treat imported sensor data. Options include:</td>
</tr>
<tr>
<td></td>
<td>• Insert and Update</td>
</tr>
<tr>
<td></td>
<td>• Insert Only</td>
</tr>
<tr>
<td></td>
<td>• Update Only</td>
</tr>
<tr>
<td>External names</td>
<td>Enter a name to tag the sensor data.</td>
</tr>
</tbody>
</table>
Cancel sensor transaction by duration
By default, this rule cancels the sensor transaction if it takes more than 20 minutes to complete.

Role required: admin
If the Discovery sensor transaction takes more than 20 minutes to complete, the following actions occur:

- The associated ECC Queue input record is set to the error state.
- The Discovery Status Completed Count is updated.
- An error message is logged, for example:

```
Sensor error when processing Linux - Storage: Transaction cancelled: maximum execution time exceeded
```

- The remainder of the discovery continues to run.

This behavior is provided by the Discovery Sensors Transaction Quota Rule. This rule is active by default.

You can disable the rule and you can modify the amount of time that has elapsed before the rule takes effect. Accessing this record requires the admin role.

1. To disable the rule, navigate to **System Definition > Quota Rules**.
   The Transaction Quotas plugin must be active.
2. Search on the name *Discovery to filter the list of rules.
3. In the name column, click Discovery Sensors.
4. To modify the amount of time, edit the Maximum Duration field of the rule.
   The value is in seconds.

Discovery probe management

Several discovery probes and their associated sensors are included with Discovery. You rarely need to modify probes or sensors. But you might need to set parameters to control the behavior of a particular probe or align versions of customized probes.

**Note:** With each release, patterns are replacing many probes and sensors for Discovery. Consider creating new patterns or editing existing ones if you want to customize what Discovery can find. The information on probes and sensors is intended for customers who are not using patterns yet and for customers who already have customized probes that are retained upon upgrade. See Patterns and horizontal discovery for more information on patterns.

What you can do

These are the things you can do with probes and sensors:

**Review the base system probes**

Review the List of Discovery probes to see the probes that exist in the base system. Probes need to be active on classifiers for Discovery to trigger the probes. However, not all probes are active on classifiers as more patterns replace probes and sensors. To see which patterns replace probes and sensors, see Pattern updates for the New York release.

**Create or modify a probe**

You can create a new probe to discover additional CIs that Discovery does not find with the base system probes or patterns, or modify an existing probe to collect additional information on
the type of CI. After you create or modify a probe, test it. You can also create multiprobes and multisensors. See Create a Discovery multiprobe for more information.

**Upgrade to the latest version of a probe or sensor**

If you upgrade your instance, the Discovery application is also updated, along with components like probes and sensors. However, if you customized any probes or sensors, they do not upgrade. You need to copy your customizations to a text file, upgrade the probes and sensors, and reapply your customizations. See Align versions of customized probes and sensors for more information.

**Set probe parameters**

Probe parameters control several aspects about how probes function. With each probe provided in the base system, certain parameters are allowed. These are specified in the list of Discovery probes. See Set probe parameters for instructions on how to set a parameter.

**Review probe permissions**

Certain probes need permissions to run on the target machines or CIs that you are trying to discover. See Discovery probe permissions for more information.

**Create or modify a probe**

Create a new probe to discover additional CIs or modify an existing probe to collect additional information. After you create or modify a probe, test it.

Role required: admin

**Important:** You need an advanced knowledge of scripting to modify probes or their associated sensors. Many existing probes provide parameters that you can set, rather than modifying the probe itself. See Set probe parameters for more information.

1. Navigate to Discovery > Discovery Definition > Probes.
2. Click the name of the probe that you want to modify.
3. Modify the form fields (see table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>Select the probe for the operating system of the device it will query.</td>
</tr>
<tr>
<td></td>
<td>- AWS Probe: Select this class to query your resources inside an AWS cloud.</td>
</tr>
<tr>
<td></td>
<td>- CIM Probe: Select this probe type to query a CIM server using WBEM protocols.</td>
</tr>
<tr>
<td></td>
<td>- Multiprobe: Select this probe type to run one or more simple probes of any type or mix probes of different types. This type of probe can make several queries simultaneously that return all the results at the same time. You cannot add multiprobes to other multiprobes.</td>
</tr>
<tr>
<td></td>
<td>- Probe: Select this generic type to define a probe class. Specify the name of the probe class in the ECC queue topic field.</td>
</tr>
<tr>
<td></td>
<td>- SNMP Probe: Select this probe type for network devices, such as routers.</td>
</tr>
<tr>
<td></td>
<td>- WMI Probe: Select this probe type for Windows devices.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| ECC queue topic     | Enter a descriptive term for the function of the probe. The probe uses this label as the **Topic** field for incoming ECC queue messages. The term does not have to be unique. For example, all the UNIX probes might have an ECC queue topic value of **SSHCommand**. By default, probes use the following ECC queue topics:  
  - CimProbe  
  - Multiprobe  
  - Powershell probe  
  - SCP Relay  
  - SSHCommand  
  - SNMP  
  - WMIRunner |
| ECC queue name      | Enter either a descriptive name for human use, or the actual command the probe is to run. For example, if the value in the ECC queue topic field is **SSHCommand**, then enter the actual shell command to run in this field. |
| Cache results       | Select this option to cache this probe’s results to improve overall discovery performance. The probe results cache should be enabled only for those probes and sensors whose output is unlikely to change. For example, the Linux – CPU sensor is cached because CPU information seldom changes. The cache is turned on by default for base system probes and sensors whose output is unlikely to change.  
  Warning: Do not turn on the cache for classification and identification probes. Furthermore, do not turn on the cache for probes that trigger additional probes because this may prevent them from being triggered. |
<p>| Classic Mode        | Select this option to cache this probe’s results in the Starting in Fuji, select this option to debug the post-processing script on the ServiceNow instance instead of the MID Server. This mode is only valid if a post-processor script exists. Use the Classic mode to debug post-processing scripts in the test environment. ServiceNow does not recommend use of the Classic mode in the production environment. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Post processor script</td>
<td>Define an optional post-processing script that runs on the MID Server. The script accepts probe results as an input and outputs a JSON string that is sent back to the instance for a sensor to use as input. Use this type of script to accomplish tasks like parsing data.</td>
</tr>
<tr>
<td></td>
<td>All probe post-processing scripts and their corresponding sensors have access to the following global variables:</td>
</tr>
<tr>
<td></td>
<td>- <strong>current</strong>: Sets the attribute to current. Same functionality that exists in sensors. They are attributes of the device that is discovered.</td>
</tr>
<tr>
<td></td>
<td>- <strong>related_data</strong>: Any additional data to be passed from the probes post-processing scripts, such as related_list.</td>
</tr>
<tr>
<td></td>
<td>All probe post-processing scripts and their corresponding sensors have access to the following methods:</td>
</tr>
<tr>
<td></td>
<td>- <strong>log(string)</strong>: Logs the string to the MID Server and the ServiceNow instance log files.</td>
</tr>
<tr>
<td></td>
<td>- <strong>getParameter(param)</strong>: Gets the parameter from the ECC queue payload.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select **Save**.
5. Modify these related lists if necessary.

<table>
<thead>
<tr>
<th>Related list</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe Parameters</td>
<td>Parameters that control the functionality of the probe. See <a href="#">Set probe parameters</a> for instructions.</td>
</tr>
<tr>
<td>Sensors that react to this probe or Sensors</td>
<td>The sensors that this probe utilizes.</td>
</tr>
<tr>
<td>Included by MultiProbe</td>
<td>If this probe is included with other probes as a MultiProbe.</td>
</tr>
<tr>
<td>MultiSensor Scripts</td>
<td>Scripts that run for multiple sensors.</td>
</tr>
<tr>
<td>Versions</td>
<td>The version of the probe. If the <strong>State</strong> is <strong>Current</strong>, the probe is the most up to date with your version of the instance. If not, you do not have the most current version and you might need to realign it. See <a href="#">Align versions of customized probes and sensors</a> for instructions.</td>
</tr>
</tbody>
</table>

6. Click **Test probe**.
7. On the Test Probe window, enter the IP address of the target and select the MID Server. Only **Up** and **Validated** MID Servers are able to be selected.
8. Click **OK**.
9. **Check the ECC Queue** for the MID Server to see the results.

Add the probe to the **Triggers Probe** related list on the appropriate classifier. See **Create a Discovery CI classification** for a description of the fields and related lists on the classifier form.

**Discovery probe permissions**
Several probes require additional permissions to run.

### Discovering Active Connections

Discovery uses the Windows - Active Connections probe to access active connection information. The **application dependency mapping** feature requires this probe to function.

<table>
<thead>
<tr>
<th>Probes</th>
<th>Additional Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - Active Connections</td>
<td>Ability to invoke the \root\CIMV2:Win32_Process object</td>
</tr>
<tr>
<td></td>
<td>Access to the admin$ share</td>
</tr>
<tr>
<td></td>
<td>Access to the %SystemRoot%\temp directory</td>
</tr>
</tbody>
</table>

### Discovering Application Profiles

Discovery uses these probes to access application profile information.

<table>
<thead>
<tr>
<th>Probes</th>
<th>Additional Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - Get APD Environment Files</td>
<td>Ability to invoke the \root\CIMV2:Win32_Process object</td>
</tr>
<tr>
<td>Windows - Get APD Env File Content</td>
<td>Access to the admin$ share</td>
</tr>
<tr>
<td>Windows - Get APD Version File Content</td>
<td>Access to the %SystemRoot%\temp directory</td>
</tr>
</tbody>
</table>

### Discovering VMware Workstation

**Attention:** The Discovery of VMware Workstation is not supported in releases after Kingston. The VMware workstation probe and sensor are not included in new instances of London. However, customers who upgrade from a release prior to Kingston can continue to use the probe and sensor to discover VMware Workstation.

Discovery uses the **Windows - Get VMware Workstation** probe to access information about VMware virtual machines installed on Windows.
Discovering VMware Workstation

<table>
<thead>
<tr>
<th>Probes</th>
<th>Additional Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - Get VMware Workstation</td>
<td>Ability to invoke the \root\CIMV2:Win32_Process object</td>
</tr>
<tr>
<td></td>
<td>Access to the admin$ share</td>
</tr>
<tr>
<td></td>
<td>Access to the %SystemRoot%\temp directory</td>
</tr>
<tr>
<td></td>
<td>Configure the PATH variable to include the path to the vmrun.exe command. This command is typically found in the VMware Workstation install directory.</td>
</tr>
</tbody>
</table>

Discovering MSSQL

Discovery uses the Windows - MSSQL probe to access information about Microsoft SQL Server installed on Windows.

<table>
<thead>
<tr>
<th>Probes</th>
<th>Additional Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - MSSQL</td>
<td>• Access to the Win32_Process object</td>
</tr>
<tr>
<td></td>
<td>• Access to the Win32_Service object</td>
</tr>
<tr>
<td></td>
<td>• Access to the Win32_NetworkAdapterConfiguration object</td>
</tr>
<tr>
<td></td>
<td>• Access to the Microsoft.SqlServer.Management.Smo object</td>
</tr>
<tr>
<td></td>
<td>• Access to run the nbtstat command</td>
</tr>
<tr>
<td></td>
<td>• Access to HKLM\SOFTWARE\Microsoft\Microsoft SQL Server\InstalledInstances</td>
</tr>
<tr>
<td></td>
<td>• Access to HKLM\SOFTWARE\Microsoft\MSSQLServer$instanceName\SuperSocketNetLib\Tcp\TcpPort where $instanceName is an array of possible values.</td>
</tr>
</tbody>
</table>

Set probe parameters

Use probe parameters to control the behavior of a particular probe every time it is triggered.

Required role: admin.

The following parameters are available for all probe types:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>(Required) The initial host to connect to.</td>
</tr>
<tr>
<td></td>
<td>• Type: string (URL)</td>
</tr>
<tr>
<td></td>
<td>• Default value: None</td>
</tr>
</tbody>
</table>

1. Create or select the **probe** you want to set parameters for.
2. From the Probe Parameters related list, click **New**.
3. Fill in the fields, as appropriate (see table).
The Probe parameter form has the following fields.

### Probe Parameter Form Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter the parameter name. Each probe type has its own list of available parameters.</td>
</tr>
<tr>
<td>Value</td>
<td>Enter the parameter value or script as required by the parameter.</td>
</tr>
<tr>
<td>Value script</td>
<td>(Optional) Enter the script you want the parameter to run if you have not already specified a script in the Value field.</td>
</tr>
<tr>
<td>Probe</td>
<td>Displays the probe this parameter relates to.</td>
</tr>
</tbody>
</table>

### Align versions of customized probes and sensors

If you customized a probe or sensor and upgraded to a new version of an instance, you need to realign the versions of the customized probe and sensor to the most current version.

Role required: admin

Probes and sensors have a major and a minor version.

**Major version**

A major version change reflects a change in the expected output of a probe, such as the addition or subtraction of a targeted attribute, a format change, such as XML versus JSON, or a probe parameter script change.

**Minor version**

A minor version change reflects small fixes that do not impact the result or the processing of the data.

A sensor and its corresponding probe must have the same major version. It is recommended they also have the same minor version. This version matching ensures that the data sent back from the probe is understood and properly processed by the sensor. All members of a multi-probes bundle must have the same major and minor version.

By default, Discovery tracks major version mismatches and displays version mismatch errors in the Active Discovery Errors section on the Discovery Dashboard and in the Discovery Log. You can control whether or not Discovery tracks minor versions mismatches by setting the Warn on Minor Version Mismatch (glide.discovery.warn_minor_version) Discovery property. Minor version mismatches are tracked in the Discovery log, but are not displayed on the Discovery Dashboard.

Versions for multi-probes and multi-sensors are checked as follows:

- The versions of the individual probes contained in the multi-probes are compared with the Responds to Probes scripts that process their data.
- The versions of the Responds to Probes scripts and the main multi-sensors script are compared.

If a probe and its corresponding sensor do not have the same major version, a sensor does not process information during a discovery and sends error messages to the log file. Errors also show up on the Discovery Dashboard when you run a discovery job. If the major version is the same, but the minor version is not, a sensor processes information during a discovery.

See the Discovery Probe and Sensor Versioning video for a tutorial on how to resolve version conflicts.
To use the most up-to-date version of a probe that is available:

1. Save the customizations that you made with your probe or sensor into a text file so you can re-apply them later.
2. Navigate to Discovery > Dashboard.
3. Look for version mismatch errors in the Active Discovery Errors section, and note the probe or sensor that caused the error.
4. Navigate to Discovery Definition > Probes or Discovery Definition > Sensors and select the probe or sensor that you want to upgrade.
5. Add the Major version and Minor version fields to the form if you want to display the versions on the form.
6. In a multi-sensor, you can verify the major version with the versions of the probes that use the sensor in the Responds to Probes related list.
7. Click the **Versions** related list.
8. Look for version where the **State** is **Previous** and the **Source** is **System upgrades**.
9. Click the date link in the **Created** column for that version to open the version record.
10. Click **Revert to this version**.
11. Navigate back to the probe or sensor form and verify that the major version is correct.
12. Reapply your customizations to the probe or the sensor.
13. Verify that your customizations are valid with the new version of the probe or sensor by running a Discovery in your test environment.

### Discovery multiprobes and multisensors

Multiprobes contain one or more simple probes configured to extract specific information from manageable devices by executing multiple queries with a single authentication.

You can schedule multiprobes to run any time in the discovery process to make exploration more efficient. One common use for multiprobes is as **identity probes**. Identity probes ask a device for information such as its name and serial number, and then use the results of those queries to update existing CIs in the CMDB.

To process the data returned from a multiprobe, you must create multisensors, which run scripts that process the data returned by the multiprobes. Multisensors typically use the same name as their corresponding multiprobes.

**Note:** The corresponding multisensors have the same names as the multiprobes.

The following multiprobes are included with the base system.

<table>
<thead>
<tr>
<th>Multiprobe</th>
<th>Includes These Probes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX - ADM</td>
<td>• AIX - Active Processes: Gets active running processes.</td>
</tr>
<tr>
<td></td>
<td>• Unix - Active Connections: Gets active connections information.</td>
</tr>
<tr>
<td>AIX - Identity</td>
<td>• AIX - Network: Determines network interfaces, IPs, and MACs.</td>
</tr>
<tr>
<td></td>
<td>• AIX - Serial Number: Gets AIX serial numbers.</td>
</tr>
<tr>
<td>CIM - Identity</td>
<td><strong>CIM - Computer System:</strong> Gets CIM Computer Systems per SMI-S.</td>
</tr>
<tr>
<td>CIM - Identity - WBEM Service</td>
<td><strong>CIM - CIMOM:</strong> Probes for the WBEM Service, CIMOM.</td>
</tr>
<tr>
<td>HP-UX - ADM</td>
<td>• HP-UX - Active Processes: Gets active running processes.</td>
</tr>
<tr>
<td></td>
<td>• Unix - Active Connections: Gets active connections information.</td>
</tr>
<tr>
<td>HP-UX - Identity</td>
<td>• HP-UX - Hardware Serial Number: Gets HP-UX serial numbers.</td>
</tr>
<tr>
<td></td>
<td>• HP-UX - Network: Determines network interfaces, IPs, and MACs.</td>
</tr>
<tr>
<td>Multiprobe</td>
<td>Includes These Probes</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Linux - Identity | * Linux - Hardware Information: Gets DMI (BIOS) information.  
* Linux - Network: Determines network interfaces, IPs, and MACs.                      |
| Mac OS X - Identity | * Mac OS X - Network: Determines network interfaces, IPs, and MACs.  
* Mac OS X - CPU/Memory: Gets CPU and memory information.                               |
| SNMP - A10 - Identity | * SNMP - A10 - Identity - Serial: Part of the SNMP Load Balancer - Identity multiprobe that is used to identify the A10 load balancer.  
* SNMP - Identity Info: Identifies an SNMP device.                                      |
| SNMP - ACE - Identity | * SNMP - Identity Info: Identifies an SNMP device.  
* SNMP - ACE - Identity - Serial: Part of the SNMP Load Balancer - Identity multiprobe, that is used to identify the ACE load balancer |
| SNMP - Alteon - Identity | * SNMP - Alteon - Identity - Serial: Part of the SNMP Load Balancer - Identity multiprobe, that is used to identify the Alteon load balancer.  
* SNMP - Identity Info: Identifies an SNMP device.                                      |
| SNMP - AppDirector - Identity | * SNMP - AppDirector - Identity - Serial: Part of the SNMP Load Balancer - Identity multiprobe, that is used to identify the AppDirector load balancer.  
* SNMP - Identity Info: Identifies an SNMP device.                                       |
| SNMP - F5 BIG-IP - Identity | * SNMP - F5 BIG-IP - Identity - Serial: Retrieves the BIG-IP chassis serial number, which is globally unique for this vendor.  
* SNMP - Identity Info: Identifies an SNMP device.                                       |
| SNMP - Identity | * SNMP - Identity Info: Identifies printer CIs                                                                                                         |
| SNMP - Load Balancer - Identity | * SNMP - F5 BIG-IP - Identity - Serial: Gets BIG-IP chassis serial numbers, which are globally unique for this vendor  
* SNMP - Identity Info: Identifies SNMP devices                                             |
<table>
<thead>
<tr>
<th>Multiprobe</th>
<th>Includes These Probes</th>
</tr>
</thead>
</table>
| SNMP - Netscaler - Identity    | • SNMP - Netscaler - Identity - Serial: Retrieves the Netscaler chassis serial number, which is globally unique for this vendor.  
• SNMP - Identity Info: Identifies SNMP devices |
| Solaris - ADM                  | • Solaris - Active Processes: Gets active running processes  
• Unix - Active Connections: Gets active connections information |
| Solaris - Identity             | • Solaris - Network: Determines network interfaces, IPs, and MACs  
• Solaris - Serial Number: Gets serial numbers for Solaris devices † |
| Unix - ADM                     | • Unix - Active Processes: Gets active running processes  
• Unix - Active Connections: Gets active connections information |
| UNIX - Classify                | • ESX - OS: Identifies ESX machines  
• UNIX - OS: Runs after the ESX - OS probe to determine the operating system for SSH devices not identified as ESX |
| Windows - ADM                  | • Windows - Active Processes: Gets active running processes  
• Windows - Active Connections: Retrieves active connections information |
| Windows - Identity             | • Windows - Network: Determines network interfaces, IPs, and MACs  
• Windows - OS/Hardware Information: Probes a Windows machine for WMI information |
• Windows - Storage 2008 - WMI: Retrieves disk and file system information for Windows 2008 and earlier. |
• Windows - Storage 2012 - PS: Retrieves disk and file system information for Windows 2012 and later. |
Note: ↑ This probe requires the installation of a command line tool from Oracle called SNEEP. To download and install this tool, log in to the Oracle website. After this tool is installed, the Solaris - Serial Number probe runs automatically when Discovery detects a Solaris device. For Fujitsu PRIMEPOWER devices, you must run this probe with root credentials.

Add a simple probe to a multiprobe
You can add simple probes to your multiprobe in the Includes Probes related list.

Role required: admin

You can add simple probes of any type to a multiprobe and mix probe types if necessary.

Warning: Do not add a multiprobe to the Includes Probes related list.

1. Navigate to Discovery Definitions > Probes.
2. Open a multiprobe from the list.
   You can see the MultiProbe designation in the ECC queue topic column.
3. In the MultiProbe record, click **New** in the **Includes probes** related list.
4. Select a simple probe from the collection list in the left column and move it into the included list.

5. Click **Save**.

*Create a Discovery multiprobe*

You can use the default multiprobes and multisensors provided in the platform or create your own.

1. Navigate to **Discovery** > **Discovery Definition** > **Probes**.
2. Click **New**.
3. Complete the **form** using the following settings:
   - Probe type: Multiprobe.
   - ECC queue topic: MultiProbe.
4. In the Includes probes related list, *add the probes* you want to include in the multiprobe.
5. Click *Save*.

*Create a Discovery multisensor*
Create a multisensor to process data returned from a multiprobe.
1. Navigate to Discovery > Discovery Definition > Sensors.
2. Click **New**.
3. Complete the form using the following settings:
   - Sensor type: MultiSensor.
   - ECC queue topic: MultiSensor.
   - Reacts to probe: Select the probe whose payload this sensor must process.
   - Sensor type: Select one of these options.
     - Javascript: Returned data from the probe is processed in the sensor itself, outside the application, and is visible to the user. This is the most common sensor type.
     - XML: The XML data from the probe is broken into pieces. Some pieces can be used to launch other probes that the original sensor needs to complete all the necessary information about a device.
4. Right-click the form header, and click **Save**.
5. In the Responds to Probes related list, click **New**.
6. Complete the form using the following settings:
   - Reacts to probe: Select the probe within the multiprobe that this sensor reacts to. The sensor must be linked to the probe by function, such as network identification or serial number.
   - Script: Enter a script to run before the script in the multisensor. This script defines how the data returned from each probe should be processed.
7. Click **Save**.

**Example custom Discovery probe and sensor: populate a CI with text file values**

This custom Discovery probe helps you if you need to read a text file from a Windows computer and populate a CI in the CMDB with the values from the file.

In this example the user wanted to read files created by BGinfo.

**Note:** When you have completed the probe and sensor, place the probe in the appropriate Windows classifier at **Discovery Definition > CI Classification > Windows**.

1. Navigate to **Discovery Definition > Probes**, and then click **New**.
2. Complete the following fields:
   - Name: Unique and descriptive name for the probe
   - Probe type: Select **Probe**.
   - Description: Describe the function of this probe.
   - Used by Discovery: Select this check box
   - ECC queue topic: This is name of the probe the MID server is to run. In this example, we use WMIRunner.
   - ECC queue name: In this example, we use the descriptive name WMI: BGInfo files.
3. Right-click in the header bar and select **Save** from the context menu.
4. Select the Probe Parameters tab in the Probe form, and then click **New**.
5. Enter **WMI_GetFiles.js** as the Name of this parameter.
6. Copy the script below into the Script field and edit as needed.
7. Click **Submit**.

```bash
//
// Use ServiceNow WMIAPI to gather stats
```
//
var CMD_RETRIES = 3;
var scanner = getScanner();
if (scanner) {
    var output = "";
    for(var i = 0; i < CMD_RETRIES; i++) {
        output = scanner.winExec("%SystemRoot%\system32\cmd.exe /C type \n"C:\\Information Systems\\BgInfo\\*.txt\\\\"");
        if (output)
            break;
    }
    scanner.appendToRoot("output", output);
}

8. Navigate to Discovery Definition > Sensors, and then click New.

Complete the following fields:
- **Name**: Use the same name as the matching probe. In this example, we use Windows - Get BGInfo files.
- **React to probe**: The name of the probe created in the previous procedure: Windows - Get BGInfo files
- **Sensor type**: Select the type of sensor to create - in this example Sensor.
- **Description**: Describes the function of this sensor.
- **Script**: Copy the script below into the Script field and edit as needed.
- **Sensor type**: Determines how the answer from the probe is processed - in this example JavaScript.

9. Click Submit.

```javascript
new DiscoverySensor({
    data: {},
    process: function(result) {
        this.parseOutput(result.output);
        this.update(this.data);
    },
    parseOutput: function(output) {
        var currentFile;
        var files = {};

        if (output.startsWith("<wmi")) {
            var bgout = new XMLHelper(output).ToObject();
            if (!bgout)
                return;
            output = bgout.output;
        }

        var lines = output.split(/
/);

        for(var i = 0; i < lines.length; i++) {
            var line = lines[i];
            if (line.startsWith("C:\Information Systems\BgInfo\") {  
                var currentFile = line.substr(30);  
                if (!files[currentFile])
                    return;
                currentFile = line.substr(30);
                files[currentFile] = ";
                } else if (currentFile) {
                    var newLine = line.trim();
                    if (newLine)
                        files[currentFile] += (files[currentFile]?
"\n" &nbsp; : "") + newLine;
```
Advanced Discovery configuration

You can configure several additional components of Discovery such as Application Dependency Mapping, the ECC queue, and extensions for the MID server.

Credential-less Discovery with Nmap

If the instance fails to identify a configuration item (CI) because of authentication failure, Discovery or Service Mapping can run selected Network Mapper (Nmap) commands with a MID Server to collect some basic information about the CI without using credentials.

A MID Server administrator can install Nmap on individual MID Servers running on a Windows host. Those MID Servers can then discover some basic information about CIs in your network when normal authentication fails.

Attention: Self-hosted customers whose network security does not permit downloads from install.service-now.com must install and configure Nmap manually on their system. Refer to Install Nmap on a self-hosted system for instructions.

Credential-less Discovery can create or modify host and application CIs when credentials are missing or misconfigured. If a credential-based Discovery is performed successfully after Nmap creates a CI, the system reconciles the information gathered from each type of discovery.

What Nmap can discover

The Nmap commands executed during credential-less Discovery can:

- Perform reverse DNS name resolution to identify the host from the IPv4 address.
- Return the MAC address of the host if that host is on the same subnet as the host executing the Nmap command.
- Detect applications installed on a target host.
- Detect the operating system of a target host and the OS version.

Note: Credential-less Discovery classifies routers and switches as hardware. It does not create or update CIs specifically for them.

Nmap credential-less Discovery scans in cloud computing platforms

It is often against the terms of service to run Nmap scans to or from any resource within a cloud computing service such as Amazon Web Service, Microsoft Azure, IBM Cloud, or Google Cloud Platform. For example, the Amazon Web Service (AWS) environment is tightly regulated and requires the permission of AWS through the AWS Vulnerability/Penetration Testing Request form. Unauthorized tests against AWS services or AWS-owned resources are prohibited. For this reason,
credential-less Discovery within a cloud computing service environment is not appropriate, and if a violation of their policy occurs, could result in expulsion from the service. Please contact your platform service provider for information on limitations or permission requirements for running Nmap.

**Components installed with Nmap**

The Discovery - IP Based (com.snc.discovery.ip_based) plugin that provides the Nmap functionality is activated automatically when either Discovery or Service Mapping is active. These Nmap components are provided by the Discovery - IP Based plugin:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System property</td>
<td>The <code>mid.discovery.credentialless.enable</code> property enables or disables Nmap for all MID Servers on which Nmap is installed that are connected to the instance. This property is installed with the Discovery plugin and is enabled by default. It is configurable by a system administrator.</td>
</tr>
<tr>
<td>MID Server properties</td>
<td>These properties, from the MID Server Property (ecc_agent_property) table, are not intended to be configured:</td>
</tr>
<tr>
<td></td>
<td>· mid.nmap.version: Version of Nmap that is installed on MID Servers in your environment. This field is visible on the MID Server (ecc_agent) form after Nmap is installed.</td>
</tr>
<tr>
<td></td>
<td>· nmap.safe.scripts: Defines the list of Nmap scripts that are classified as safe for use during execution of Nmap’s Application Version Detection phase (-sV command option).</td>
</tr>
<tr>
<td></td>
<td>· nmap.npcap.version: The version of Npcap that is installed with Nmap. The Nmap installer can only perform upgrades of existing Npcap installations it encounters.</td>
</tr>
<tr>
<td>Fields</td>
<td>· Credentialless Discovery Port (cl_port): Optional field on the Application (cmdb_ci_appl) table that displays the number of a port scanned by credential-less Discovery. This port number is used to determine whether an application returned by Nmap has a matching CI in the CMDB or if a new CI must be created.</td>
</tr>
<tr>
<td></td>
<td>· Discovery source (discovery_source): Optional field in the Configuration Item (cmdb_ci) table to which the CredentiallessDiscovery choice is added. This option shows that credential-less Discovery was used to create a CI.</td>
</tr>
<tr>
<td>Component</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Nmap MID Server capability</td>
<td>The <strong>Nmap capability</strong> is added to the MID Server when Nmap is installed and removed automatically when Nmap is uninstalled. Only MID Servers with this capability can perform credential-less Discovery. A system administrator cannot add or remove this capability manually. Self-hosted customers who have the maint role can modify or delete the Nmap capability, but <em>should not do so</em>. Service Mapping does not check for the presence of the <strong>Nmap capability</strong> and selects the MID Server based on the IP address only. To ensure that Service Mapping does not select a MID Server without the <strong>Nmap capability</strong>, install Nmap on all MID Servers assigned to the IP address ranges for which you want credential-less Discovery to be available. If Service Mapping selects a MID Server for credential-less Discovery that does not have Nmap capabilities, this error message appears in the map, at the site of the CI being discovered: <em>Nmap is not installed on MID Server. Verify all MIDs configured to handle selected IP Address have Nmap Capability. Nmap root directory path does not exist:</em> &lt;path&gt;</td>
</tr>
<tr>
<td>Npcap</td>
<td><strong>Npcap</strong> is Nmap©’s packet capture library for Windows. Npcap allows Nmap to perform port scans quickly and to identify the family of the operating system running on the target. Only one copy of Npcap is installed per MID Server host. Because Npcap can be used by other applications, uninstalling Nmap does not automatically uninstall Npcap. You must uninstall Npcap manually, after determining that no other dependencies exist.</td>
</tr>
</tbody>
</table>

**Note:** The ALL MID Server capability does not include the **Nmap capability**.
<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
</table>
| Patterns  | • **Credentialless Discovery Network Device**: Scans a host IP address using an Nmap command to identify the host. This pattern launches the **Credentialless Discovery Network Device – PreLaunch** script to retrieve the list of ports to explore from the IP Service (cmdb_ip_service) table. Do not modify this script.  
  • **Credentialless Discovery Application**: Scans a port at an IP address using an Nmap command to identify the application service actively listening on that port. Service Mapping launches this pattern when all credential-based port classification steps fail. Discovery creates a CI in the Application (cmdb_ci_appl) table if the port is open and it can identify the service by name and product. If the service does not respond to any of the scan attempts, Nmap consults its nmap-services registry and guesses at which service is most likely running on that port. If Nmap has to guess what application is running on a scanned port, the Credentialless Discovery Application pattern does not create an application CI or update an existing CI. |
| MID Server script includes | • **SetCredentialLessDeviceClassName**: Determines which host CI to create or update after the successful execution of the Nmap command. Do not modify this script.  
  • **CredentialLessApplicationClassNameMapper**: Maps the service product, service name, and extra service information supplied by Nmap for the scanned port to a supported application table in the instance. System administrators can modify this script.  
  • **SetCredentialLessApplicationClassName**: Ensures that the CredentialLessApplicationClassNameMapper script is invoked only once. Do not modify this script. |
| System script include | The **CredentiallessDiscoveryAjax** script include runs on the instance and handles the installation and uninstallation of Nmap on Windows MID Servers, executed from UI actions on the form. Do not modify this script. |

**Credential-less host Discovery**

Credential-less host discovery occurs when a scanned host is found to be alive, but not active, or when all configured credential-based classification probes have failed.
How the host Discovery pattern is launched

If the Shazzam probe scans a host that is alive but not active, and if credential-based classification probes fail, horizontal Discovery launches the Credentialless Discovery Network Device pattern to gather host information. If the host being scanned does not have a CI defined, Service Mapping launches the HorizontalDiscoveryProbe probe, which in turn launches the Credentialless Discovery Network Device pattern. This pattern attempts to create a new CI if one does not already exist for the scanned host or to update an existing CI in the Hardware (cmdb_ci_hardware) table.

**Important:** To allow the Credentialless Discovery Network Device pattern to launch, ensure that the `mid.discovery.credentialless.enable` system property is set to `true`.

ECC Queue entries

The system creates these entries in the ECC queue during execution of the HorizontalDiscoveryProbe.
Log messages

These log messages are published during execution of the HorizontalDiscoveryProbe.

The Nmap command

The Nmap command executed on the MID Server determines if the host is up. Using the IP address and a list of ports passed to Nmap by the pattern, Nmap performs reverse DNS name resolution to identify the host name associated with the IP address. If multiple host names are configured, the first name returned by Nmap is used. If no host name is configured, then the IP address of the remote host is used to create the host CI. The Npcap packet capture library, included with the Nmap installation, identifies the host operating system family. If the scanned host is located on the same subnet as the Windows MID Server host that executes Nmap, the remote host's MAC address is returned.

Note: The list of ports that Nmap is configured to scan during credential-less host Discovery is stored in the IP Service (cmdb_ip_service) table, which is editable. By default, all ports are available for scanning. To block the use of any port for an Nmap scan, set the value in the Credentialless Discovery (cl_discovery) column to false.
Creating or updating host CIs

After successful execution of the Credentialless Discovery Network Device pattern, the SetCredentialLessDeviceClassName MID Server script runs to identify the operating system family of the discovered host. The system class of the host CI is based on the operating system family that is returned by Nmap. If the OS family matches one of the six supported server operating systems, then the system uses a server class derived from the Hardware (cmdb_ci_hardware) base class, such as Linux Server (cmdb_ci_linux_server). If no match is found, the system uses the base class.

**Note:** When the issue with the credentials is resolved and Discovery runs again, the instance uses the serial number, host name, and system class provided by credential-based discovery to update the host CI that was created by credential-less Discovery.

### Supported OS families

<table>
<thead>
<tr>
<th>OS family</th>
<th>CI table</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX</td>
<td>cmdb_ci_aix_server</td>
</tr>
<tr>
<td>HP-UX</td>
<td>cmdb_ci_hpux_server</td>
</tr>
<tr>
<td>Linux</td>
<td>cmdb_ci_linux_server</td>
</tr>
<tr>
<td>Solaris</td>
<td>cmdb_ci_solaris_server</td>
</tr>
<tr>
<td>OS X or iOS</td>
<td>cmdb_ci_osx_server</td>
</tr>
<tr>
<td>Windows</td>
<td>cmdb_ci_win_server</td>
</tr>
<tr>
<td>Undefined</td>
<td>cmdb_ci_hardware</td>
</tr>
</tbody>
</table>
Hardware identification

The Discovery - IP Based (com.snc.discovery.ip_based) plugin adds an identifier to the Hardware Rule for the Hardware (cmdb_ci_hardware) base table that matches on the host CI name for Nmap scans. The Hardware Rule is used by both credential-based and credential-less Discovery.

Credential-less Application Discovery

Credential-less Application Discovery attempts to identify an application service actively listening on a specific port at a given IP address.

The application discovery pattern

Service Mapping launches the Credentialless Discovery Application pattern when all credential-based port classification steps fail. The pattern executes an Nmap command on a Windows MID Server with Nmap installed, that is configured to perform application/version detection against a specific remote host IP address and port. If the port being scanned by Nmap is open, the pattern executes the CredentialLessApplicationClassNameMapper MID Server script include, which maps the service product, service name, and any extra information supplied by Nmap to a supported ServiceNow application table. If the script can map the returned product to an appropriate table derived from the base Application (cmdb_ci_appl) table, the script passes this information to the pattern. The pattern passes the match to the Discovery identifier for eventual CI creation or reconciliation. If the information returned by Nmap does not match any derived table, then the instance uses the base Application (cmdb_ci_appl) table to create the CI.

Important: To allow the Credentialless Discovery Application pattern to launch, ensure that the mid.discovery.credentialless.enable system property is set to true.
**Example scan**

This information was returned by an Nmap Application/Version Detection port scan on a Linux test system and illustrates the type of application data Nmap scans can return.

<table>
<thead>
<tr>
<th>Information</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port state</td>
<td>open</td>
</tr>
<tr>
<td>Service name</td>
<td>ssh</td>
</tr>
<tr>
<td>Service product</td>
<td>OpenSSH</td>
</tr>
<tr>
<td>Extra service information</td>
<td>Protocol 2.0</td>
</tr>
</tbody>
</table>

**Default application mappings**

The `CredentialLessApplicationClassNameMapper` MID Server script include is configured with a subset of the most common application tables available for Discovery and Service Mapping. A user with the agent_admin role can edit this script include to add additional CI tables that credential-less application Discovery can use for mapping to a derived application CI class.

Shown in this table are examples of close matches returned by Nmap on a test system that `CMDB Identification and Reconciliation` was able to resolve into defined products. In many cases, the service name returned by Nmap was not needed to determine a match. Service names that appear in the table were required to determine a match.

Products returned by Nmap that cannot be resolved into defined Discovery products use the following naming format: `<serviceProduct>:<serviceExtrainfo>:<serviceName>`. If any value in this string is null, it is dropped from the name.

- **serviceProduct**: Service product information returned by Nmap.
- **serviceExtrainfo**: Any additional information that Nmap returns about the application that might help identify it, such as protocol information.
- **serviceName**: The installed name of the service or daemon of the product.

**Examples of close matches that were resolved**

<table>
<thead>
<tr>
<th>Service product</th>
<th>Service name</th>
<th>Service extra information</th>
<th>Discovered product</th>
<th>CI application table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Tomcat/ Coyote JSP engine 1.1</td>
<td>N/A</td>
<td>NULL</td>
<td>Tomcat</td>
<td>cmdb_ci_app_server_tomcat</td>
</tr>
<tr>
<td>Apache httpd 2.2.10 ((Linux/SUSE))</td>
<td>N/A</td>
<td>NULL</td>
<td>Apache Web Server</td>
<td>cmdb_ci_apache_web_server</td>
</tr>
<tr>
<td>IBM HTTP Server</td>
<td>N/A</td>
<td>Derived from Apache</td>
<td>Apache Web Server</td>
<td>cmdb_ci_apache_web_server</td>
</tr>
<tr>
<td>IBM DB2 Database Server (QDB2/LINUX)</td>
<td>N/A</td>
<td>NULL</td>
<td>DB2 Instance</td>
<td>cmdb_ci_db_db2_instance</td>
</tr>
<tr>
<td>Microsoft Exchange smtpd</td>
<td>smtp</td>
<td>NULL</td>
<td>Exchange Client Access Server</td>
<td>cmdb_ci_exchange_cas</td>
</tr>
</tbody>
</table>
### Examples of applications not uniquely matched

In this example, the information returned by Nmap does not match any derived table, and the instance must use the base Application (cmdb_ci_appl) table to create the CI.

<table>
<thead>
<tr>
<th>Scanned application</th>
<th>Nmap response</th>
<th>Script response - Identifier input</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExchangeHub</td>
<td>Microsoft Windows RPC</td>
<td>msqlrpc</td>
</tr>
<tr>
<td>Scanned application</td>
<td>Nmap response</td>
<td>Script response - identifier input</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td></td>
<td>Service product</td>
<td>Service name</td>
</tr>
<tr>
<td>HAProxy Load Balancer</td>
<td>IBM HTTP Server (Derived from Apache)</td>
<td>http</td>
</tr>
<tr>
<td>SharePoint</td>
<td>Oracle Database</td>
<td>http</td>
</tr>
<tr>
<td>SharePoint</td>
<td>Oracle Instance</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Application identification**

The Discovery - IP Based (com.snc.discovery.ip_based) plugin adds an identifier to the Application Rule for the Application (cmdb_ci_appl) table that matches on sys_class_name and cl_port for Nmap scans.
<table>
<thead>
<tr>
<th>Identifier</th>
<th>Application Rule</th>
<th>Active</th>
<th>Applies to</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifier Entries (3) Related Entries</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier Entries</td>
<td>Priority</td>
<td>Search</td>
<td></td>
<td></td>
</tr>
<tr>
<td>identifier = cmdb_ci_app</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Active</td>
<td>Search on table</td>
<td>Criterion attributes</td>
<td>Allow null attribute</td>
</tr>
<tr>
<td>1</td>
<td>true</td>
<td>Application [cmdb_ci_app]</td>
<td>sys_class_name,cl_port</td>
<td>false</td>
</tr>
<tr>
<td>2</td>
<td>true</td>
<td>Application [cmdb_ci_app]</td>
<td>running_process_command,running_process_</td>
<td>true</td>
</tr>
<tr>
<td>3</td>
<td>false</td>
<td>Application [cmdb_ci_app]</td>
<td>sys_class_name,p_command_hash,p_key_pass</td>
<td>true</td>
</tr>
</tbody>
</table>

Actions on selected rows...
Nmap commands and data collected with credential-less Discovery
Nmap executes in phases when collecting data and runs a controlled set of safe commands with two patterns for exploring applications and devices.

Data returned by Nmap from host Discovery
Nmap returns the following data for a host it determines is up:

- **Operating system family**: If the operating system family is one of the six supported server operating system families, then a host CI associated with the appropriate server class is used to create or update the host CI. If the operating system family returned by Nmap is not one of the supported server operating systems, then a host CI is created or updated using the base Hardware (cmdb_ci_hardware) class.

- **MAC address**: If the scanned host is located on the same subnet as the Windows MID Server host that is executing the Nmap commands, then Nmap returns the remote host's MAC address.

Commands run by Nmap during host Discovery
The MID Server passes the IP address of a remote host to the Nmap command and a set of ports, taken from the IP Service table. The host must be reported by Nmap to be up. The host is up if the port state returned by Nmap is **open** or **open/filtered**. If the host is reported to be up, then the operating system family returned is examined to determine the class used to create or update the host CI.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| -PS     | TCP SYN option determines if the host is online. This option that sends an empty, raw TCP packet with the SYN flag set to all the specified ports on the scan list. Nmap reports the host is online if a SYN/ACK TCP packet is returned and if an RST is returned, indicating that the port is closed. The Nmap host machine tears down the nascent connection with an RST, instead of an ACK, since the SYN/ACK response is unexpected from its perspective. If multiple port probes are specified, they are sent in parallel. Requirements:  
  - Raw packet privileges.  
  - Must run as root on UNIX.  
  - Administrator privileges on Windows. |
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| -PA     | Detects packet-filtering devices. This option is a TCP ACK host Discovery, using a specified list of ports. If no list is defined, the default port 80 is used. Nmap sends an empty, raw TCP packet with the ACK flag set. This action acknowledges data over an established TCP connection, causing the remote host to respond with an RST packet. If the remote host does respond with an RST packet, Nmap knows the host is online. The ACK ping probe may get past packet filtering of TCP SYN packets by stateless packet filtering, network entities, such as firewalls, routers, and host routing tables/firewall software. Stateful firewalls drop unexpected packets, causing the ACK packet to be dropped, which increases the likelihood of the SYN TCP probe getting through. Requirements:  
  - Raw packet privileges.  
  - Must run as root on UNIX.  
  - Administrator privileges on Windows.  

  **Note:** Both Ping SYN (-PS) and Ping ACK (-PA) packets types can be configured together in the same scan attempt. |
| -O      | Enables OS detection against a remote host providing the vendor name, underlying OS, OS generation, and device type, such as a router or a switch.  
  Nmap uses TCP/IP stack fingerprinting to determine the OS family running on the host. Nmap sends a series of TCP and UDP packets to the host and examines the responses. After testing the responses thoroughly, Nmap compares results to its `nmap-os-db` database and prints the OS details, if there is a match. If Nmap is not able to guess the OS, and at least one open port and one closed port were found, then it generates a fingerprint and a URL to submit to `nmap.org`.  
  OS detection does not use scripts from the Nmap Scripting Engine (NSE). OS detection runs after Application/Service Version detection, which helps the accuracy of OS Detection. However, the results of the two scan types might be contradictory. OS Detection runs at the TCP transport layer, while Application/Service Version Detection runs at a higher level in the TCP stack.  
  Requirements:  
  - Raw packet privileges.  
  - Must run as root on UNIX.  
  - Administrator privileges on Windows. |
<p>| -p      | List of ports to scan. |
| -T4     | Nmap aggressive timing template. |
| -v      | Enable verbose detail. This command is normally used to provide progress indication to viewers of standard output. However, it can cause the Nmap Scripting Engine (NSE) scripts to provide additional detail. |
| -Pn     | Disable the default ping. This command skips the initial Nmap host discovery scan phase and treats all hosts as if they are online. Use this command if you want a port scan even if the set of ports run by the default Discovery probes will not find an open port. |
| -r      | Scan ports consecutively, without randomizing. |
| --reason | Explains why Nmap classified a port with a specific state. |</p>
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--system-dns</td>
<td>Use the DNS resolver of the operating system for reverse DNS lookup, rather than the DNS name servers. This option is slower than using DNS servers, since it can only resolve one IP address at a time, while multiple rDNS requests are performed in parallel. However, the value returned matches the value returned by the MID Server that is used to create host CIs.</td>
</tr>
<tr>
<td>-X -</td>
<td>Redirects stdout to a XML file.</td>
</tr>
<tr>
<td>2&gt; &lt;file-path&gt;</td>
<td>Redirects errors messages written by Nmap to stderr to the specified file.</td>
</tr>
</tbody>
</table>

**Data returned by Nmap during application Discovery**

Nmap returns the following application data from a host it determines is up:

- **Port state:**
  - **open**: An application is actively accepting TCP (or UDP) packets on the port.
  - **open|filtered**: Nmap cannot determine if the port is open or filtered. An open port may not return a response when using Nmap’s UDP, IP protocol, FIN, NULL and Xmas scans.

- **Service name**: Name of the service actively accepting TCP (or UDP) packets on the service port.

- **Service product**: Name of the application running on the open port, as identified by Nmap.

- **Extra service information**: Any additional information returned by Nmap that Discovery can use to identify the application.

**Commands run by Nmap during application Discovery**

The MID Server passes one IP address for a remote host to the Nmap command and one port to scan. Nmap attempts to identify the application actively listening on the port and then determines whether it should create a new CI or update an existing one. The host must be reported by Nmap to be up.

Nmap uses this information to determine which course to take:

- Service product
- Service name
- Extra service information
- State of the scanned port

**Note**: The service product and service name values cannot be NULL, and the service name cannot end with a question mark. Nmap appends a question mark to a service name when it guesses at the application listening on the scanned port. The instance does not create or update an application CI if Nmap cannot retrieve the application name from its static nmap-services registry file.
**Command** | **Description**
---|---
-sS | TCP SYN (stealth) port scan. This is the default scan option and never completes TCP connections. Scans can differentiate between ports that are open, closed, or filtered.

TCP SYN scan implements a half-open reset that aborts the request before the connection can be established. The port is open (the application is listening) if a SYN/ACK is returned. The port is closed if an RST is returned. If no response is received after multiple retransmissions, the port is marked filtered. The port is also marked filtered if an ICMP unreachable error (type 3, code 1,2,3,9,10,13) is received.

Requirements:
- Raw packet privileges
- Must run as root
- Windows administrator privileges

-sV | Probes open ports to determine the service protocol, application name, version number, hostname, device type, OS Family and miscellaneous details such as the SSH protocol version.

When Nmap receives a response it cannot match to the database it prints out a fingerprint and URL for submission to nmap.org. It looks for the port number in its static nmap-services registry file to determine the name of an application service that might be actively listening on the port.

The service name returned by Nmap is appended with a question mark, indicating that Nmap had to guess at its identity. The instance does not create or update an application CI when Nmap guesses at the application running on an open port.

**Note:** This command does not require raw-packet privileges.

--datadir | Path to the directory containing the safe set of Nmap scripts to be run during Application/Version detection.

-p | Single port to scan.

-T4 | Nmap aggressive timing template.

-v | Enable verbose detail. This command is normally used to provide progress indication to viewers of standard output. However, it can cause the Nmap Scripting Engine (NSE) scripts to provide additional detail.

-Pn | Disable the default ping. This command skips the initial Nmap host discovery scan phase and treats all hosts as if they are online. Use this command if you want a port scan even if the set of ports run by the default Discovery probes will not find an open port.

-r | Scan ports consecutively, without randomizing.

--reason | Explains why Nmap classified a port with a specific state.

--system-dns | Use the DNS resolver of the operating system for reverse DNS lookup, rather than the DNS name servers. This option is slower than using DNS servers, since it can only resolve one IP address at a time, while multiple rDNS requests are performed in parallel. However, the value returned matches the value returned by the MID Server that is used to create host CIs.

-X | Redirects stdout to a XML file.

2> <file-path> | Redirects errors messages written by Nmap to stderr to the specified file.
PowerShell for Discovery and Service Mapping

MID Servers use PowerShell and PowerShell Remoting for accessing configuration items (CIs) during horizontal and top-down discovery. Review MID Server parameters and script includes, probe parameters, and credentials for using PowerShell.

PowerShell is used to control and automate the administration of Windows servers and applications.

MID Servers can use PowerShell to directly communicate with Windows servers using both WMI and WinRM protocols. For Windows services using the WinRM protocol, the PowerShell process establishes a secure PSSession (PowerShell Remoting session) that stays open until the MID Server finishes querying a Windows server. For Windows servers using the WMI protocol, the PowerShell process sends every PowerShell command with credentials.

As default, MID Servers use a WMI (Windows Management Instrumentation) Collector service that helps MID Servers to communicate with Windows servers. Patterns used to discover Windows servers or applications running on them, contain WMI and WinRM queries and commands to run on Windows servers. A WMI Collector service transfers WMI and WinRM queries and commands from the MID Server to Windows-based CIs and brings the results of the queries to the MID Server.

PowerShell is also the preferred method for performing discovery over multiple Windows domains. PowerShell allows a single MID Server to authenticate on servers on different domains using credentials stored on the instance.

If you do not configure MID Servers to use PowerShell and PowerShell Remoting, MID Servers use WMI Collector.

How PowerShell Discovery Works

The following descriptions explain how MID Servers use PowerShell to deploy probes.

Probe and sensor

When a Windows machine is classified with PowerShell, and an MSSQL instance is detected, a probe called Windows - MSSQL is launched. The probe returns the SQL database catalogs and version to a matching sensor.

Probe parameter

The WMI_ActiveConnections.ps1 probe parameter contains a script that runs netstat.exe on a target server when PowerShell is enabled. This script extracts the information on Windows server connections, such as process IDs, ports, and IP addresses.

Credentials

Discovery uses Windows PowerShell credentials from the Credentials (discovery_credentials) table or the domain administrator credentials of the MID Server service. If Discovery cannot find PowerShell credentials in the Credentials table of the type, Windows), it uses the login credentials of the MID Server service.

MID Server Script Includes

The following script includes were added for PowerShell discoveries. These scripts run on the MID Server to generate the scripts that Discovery uses for WMIRunner and PowerShell.

- GenerateWMIScriptJS: Generates a Javascript script for the WMIRunner probe.
- GenerateWMIScriptPS1: Generates a PowerShell script for PowerShell discovery.
MID Server parameters for PowerShell

Optional parameters for the MID Server can be found at [MID Server parameters for PowerShell](#). After changing the setting for any parameter, be sure to restart the MID Server service.

PowerShell version requirements

MID Servers using PowerShell must be installed on a supported Windows operating system. ServiceNow supports these PowerShell versions:

**Version 2.0**
- Regular Discovery

**Version 3.0**
- Regular Discovery
- Application Dependency Mapping (ADM)
- File-based Discovery
- PowerShell version 3.0 does not support Windows Server 2003.

**Version 4.0**
- Regular Discovery
- Application Dependency Mapping (ADM)
- File-based Discovery

**Version 5.0**
- Regular Discovery
- Application Dependency Mapping (ADM)
- File-based Discovery

**Note:** PowerShell version 6.0 is **not** supported. Many of the cmdlets that discovery relies on have been removed from this version. For example, only cmdlets using WinRM are available for remote operations.

Windows PowerShell execution policies

Windows PowerShell has four different execution policies. Customers can set the script execution policies with their group policy settings. For more information, see the Microsoft website for [PowerShell](#) documentation.

- **Restricted:** No scripts can be run. Windows PowerShell can be used only in interactive mode.
- **AllSigned:** Only scripts signed by a trusted publisher can be run.
- **RemoteSigned:** Downloaded scripts must be signed by a trusted publisher before they can be run.
- **Unrestricted:** No restrictions; all scripts can be run.

**Note:** If you have any policy other than Unrestricted, the script needs to be signed.
Set up MID Servers to use PowerShell
Configure MID Servers in your organization to use PowerShell for horizontal and top-down discovery of Windows servers.

Role required: admin

1. Download PowerShell and install it on each MID Server configured to discover Windows computers.
2. Enable PowerShell-based discovery by setting the `mid.use_powershell` MID Server parameter to true on all the MID Servers.
3. Enable discovery using PowerShell Remoting by setting the `mid.sa.prefer_powershell` MID Server property to true on all the MID Servers.
4. Determine which credentials MID Servers use for PowerShell.
   - To discover Windows computers using credentials from the credentials table, set the `mid.powershell.use_credentials` parameter to true. Using credentials form the credentials table is the default behavior when PowerShell is enabled.
   - To force Discovery and Service Mapping to use the credentials of the MID Server service user, set the `mid.powershell.use_credentials` parameter to false on the MID Server. The MID Server service must have domain admin credentials to have access to the Windows machines in the domain.
5. Optional: If necessary, configure advanced parameters for PowerShell and PowerShell Remoting on each MID Server separately.

PowerShell remoting for Discovery
Probe developers can use the PowerShell remote execution framework to automatically handle remote execution of scripts on target devices. The unified framework removes inconsistencies in remote execution, increases efficiency, and improves stability.

Requirements
To use PowerShell remote execution framework, the following requirements must be met:
- The MID Server must be able to write to and read from the network share of the target.
- The remote target must have PowerShell 2 or higher.
- For MID Servers using WinRM or WMI which choose to copy the script to the remote target, the `MachinePolicy` and `UserPolicy` scopes must be set to Undefined. If the script is not copied, the execution policy can be any other setting up to Restrictive.

Application discovery
The PowerShell remote execution framework has options for copying files to the remote target when executing a scan. Copying files to the target is important for probes like Windows — File discovery because its script invokes itself on the remote target to spawn off a new process. MID Servers using WMI to execute scripts remotely can experience an error with launchProcess if the script is too long. Copying the script to the remote target resolves this error. Copying a script to a remote target may result in anti-virus software on the target flagging the script. To avoid issues with anti-virus software, add the scripts to the permitted list in the anti-virus app.
Probe configuration

The following is the configuration page for the Windows — Active Connections probe, which is included in the Windows — ADM multiprobe.

The **Execute script remotely** check box is visible when the ECC queue topic is WMIRunner or PowerShell. When checked, the script executes on the remote target. Otherwise the script executes on the MID Server.

The **Copy script to target** check box is visible when **Execute script remotely** is checked. If **Copy script to target** is checked, then the script is copied to, and run on, the target. If **Copy script to target** is unchecked, then the script is executed on the target without copying it.

PowerShell probe development

PowerShell remote execution framework is a unified method of executing PowerShell scripts, contained in a probe parameter, on a remote target server. The framework removes the need for probe developers to write their own remote execution code, which can cause inconsistencies between developers. The probe developer writes the script as if the probe was collecting
information locally because the remote execution framework automatically handles remote script execution.

The framework handles remote execution whether the MID Server is configured to use WMI or WinRM. If the MID Server is configured for WMI, the probe uses launchProcess to execute commands on the remote target. Using launchProcess complicates remote execution and can cause failures. However, a MID Server configured for WinRM does not use launchProcess, and so is more efficient and stable.

**PowerShell commands run by Discovery**

These are the PowerShell cmdlets and their parameters that Discovery runs to control and automate the administration of Windows servers and applications. Included are the probe parameters and MID Server scripts that contain the commands.

<table>
<thead>
<tr>
<th>Cmdlet/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Add-Content</strong></td>
<td>Append content to a specified item or file. You can specify the content by typing the content in the command or by specifying an object that contains the content.</td>
</tr>
<tr>
<td>Path</td>
<td>Probe Windows - File discovery (filebaseddiscovery.ps1)</td>
</tr>
<tr>
<td>Encoding</td>
<td>MID Server script CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td>Value</td>
<td></td>
</tr>
</tbody>
</table>

| **Add-Member**    | Lets you add properties and methods to an instance of a Windows PowerShell object. For instance, you can add a NoteProperty member that contains a description of the object or a ScriptMethod member that runs a script to change the object. |
| NotePropertyName  | Probe Windows - File discovery (filebaseddiscovery.ps1) |
| NotePropertyValue | MID Server script CollectConnectionsAndProcessesInfo.ps1 |
| MemberType        | |
| Name              | |
| Value             | |

<p>| <strong>Add-Type</strong>      | Adds a Microsoft .NET Framework class in your Windows PowerShell session. You can then use the New-Object cmdlet to instantiate objects and use those objects, just as you would use any .NET Framework object. If you add an Add-Type command to your Windows PowerShell profile, the class is available in all Windows PowerShell sessions. |
| TypeDefinition    | Probes Windows - MSSQL (sqlinfo.ps1) |
| AssemblyName      | Windows - Azure (is.azure.ps1) |
| MemberDefinition  | MID Server script |
| Name              | Credentials.psm1 |
| Namespace         | RegistryAPI.psm1 |
| PassThru          | LaunchProc.psm1 |</p>
<table>
<thead>
<tr>
<th>Cmdlet/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compare-Object</td>
<td>Compares two sets of objects. One set of objects is the reference set, and the other set is the difference set. The result of the comparison indicates whether a property value appeared only in the object from the reference set or, if the IncludeEqual parameter is specified, in both objects.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>DiagnosticsUtil.psm1</td>
</tr>
<tr>
<td>ConvertTo-SecureString</td>
<td>Converts encrypted standard strings into secure strings. It can also convert plain text to secure strings. It is used with ConvertFrom-SecureString and Read-Host. The secure string created by the cmdlet can be used with cmdlets or functions that require a parameter of type SecureString.</td>
</tr>
<tr>
<td></td>
<td><strong>Probes</strong></td>
</tr>
<tr>
<td></td>
<td>Windows - Storage 2012 - PS (storage.2012.psm1)</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>Credentials.psm1</td>
</tr>
<tr>
<td>ConvertTo-Json</td>
<td>Converts any object to a string in JavaScript Object Notation (JSON) format. The properties are converted to field names, the field values are converted to property values, and the methods are removed.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>CollectConnectionsAndProcessesInfo.psm1</td>
</tr>
<tr>
<td>Copy-Item</td>
<td>Copies an item from one location to another location in the same namespace. For instance, it can copy a file to a folder, but it cannot copy a file to a certificate drive.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td></td>
<td>ExecuteRemote.psm1</td>
</tr>
<tr>
<td></td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td></td>
<td>NmapInstallation.psm1</td>
</tr>
<tr>
<td>ForEach-Object</td>
<td>Performs an operation against each item in a collection of input objects. The input objects can be piped to the cmdlet or specified by using the InputObject parameter.</td>
</tr>
<tr>
<td></td>
<td><strong>Probes</strong></td>
</tr>
<tr>
<td></td>
<td>Windows - JBoss Find web.xml List (indwebxml.psm1)</td>
</tr>
<tr>
<td></td>
<td>Windows - MSSQL (sqlinfo.psm1)</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>CollectConnectionsAndProcessesInfo.psm1</td>
</tr>
<tr>
<td></td>
<td>DiagnosticsUtil.psm1</td>
</tr>
<tr>
<td></td>
<td>PSScript.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Format-List</td>
<td>Formats the output of a command as a list of properties in which each property is displayed on a separate line. You can use Format-List to format and display all or selected properties of an object as a list.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>RegistryAPI.psm1</td>
</tr>
<tr>
<td>Get-ChildItem</td>
<td>Gets the items in one or more specified locations. If the item is a container, it gets the items inside the container, known as child items. You can use the Recurse parameter to get items in all child containers.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>CollectConnectionsAndProcessesInfo.psm1, DiagnosticsUtil.psm1, ExecuteRemote.psm1, LaunchProc.psm1, NmapInstallation.psm1, PSScript.psm1, RegistryAPI.psm1</td>
</tr>
<tr>
<td>Get-Content</td>
<td>Gets the content of the item at the location specified by the path, such as the text in a file. It reads the content one line at a time and returns a collection of objects, each of which represents a line of content.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - ADM Enhanced</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>ADMELauncherWMI.psm1, ADMELauncherWinRM.psm1, ScriptAPI.psm1</td>
</tr>
<tr>
<td>Get-Date</td>
<td>Gets a DateTimeOffset object that represents the current date or a date that you specify. It can format the date and time in several Windows and UNIX formats. You can use Get-Date to generate a date or time character string, and then send the string to other cmdlets or programs.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - File Discovery (filebaseddiscovery.psm1)</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>DiagnosticsUtil.psm1, LaunchProc.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Get-Item</td>
<td>Gets the item at the specified location. It does not get the contents of the item at the location unless you use a wildcard character (*) to request all the contents of the item.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - ADM Enhanced</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>ExecuteRemote.psm1</td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWinRM.psm1</td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td>Get-ItemProperty</td>
<td>Gets the properties of the specified items. For example, you can use this cmdlet to get the value of the LastAccessTime property of a file object. You can also use this cmdlet to view registry entries and their values.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - ADM Enhanced</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>RegistryAPI.psm1</td>
</tr>
<tr>
<td></td>
<td>ScriptAPI.psm1</td>
</tr>
<tr>
<td></td>
<td>NmapInstallation.psm1</td>
</tr>
<tr>
<td>Get-Member</td>
<td>Gets the members, the properties and methods, of objects. To specify the object, use the InputObject parameter or pipe an object to Get-Member. To get information about static members, the members of the class, not of the instance, use the Static parameter. To get only certain types of members, such as NoteProperties, use the MemberType parameter.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - ADM Enhanced</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>CollectConnectionsAndProcessesInfo.psm1</td>
</tr>
<tr>
<td>Get-Process</td>
<td>Gets the processes on a local or remote computer. Without parameters, this cmdlet gets all of the processes on the local computer. You can also specify a particular process by process name or process ID (PID) or pass a process object through the pipeline to this cmdlet.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - File Discovery (filebaseddiscovery.psm1)</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>CollectConnectionsAndProcessesInfo.psm1</td>
</tr>
<tr>
<td></td>
<td>Get-PEB.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Get-PSDrive       | Gets the drives in the current session. You can get a particular drive or all drives in the session.  
  **Probes**  
  Windows - ADM Enhanced  
  **MID Server script**  
  - ADMELauncherWMI.psm1  
  - Credentials.psm1 |
| Get-PSSession     | Gets the user-managed Windows PowerShell sessions (PSSessions) on local and remote computers.  
  **MID Server script**  
  Credentials.psm1 |
| Get-Variable      | Gets the Windows PowerShell variables in the current console. You can retrieve just the values of the variables by specifying the ValueOnly parameter, and you can filter the variables returned by name.  
  **MID Server script**  
  - ExecuteRemote.psm1  
  - PSScript.ps1  
  - DiagnosticsUtil.psm1 |
| Get-WmiObject      | Gets instances of Windows Management Instrumentation (WMI) classes or information about the available WMI classes. To specify a remote computer, use the ComputerName parameter. If the List parameter is specified, the cmdlet gets information about the WMI classes that are available in a specified namespace. If the Query parameter is specified, the cmdlet runs a WMI query language (WQL) statement.  
  **Probes**  
  - Windows - MSSQL (sqlinfo.ps1)  
  - Windows - Storage 2008 - PS (storage.ps1)  
  **MID Server script**  
  - ExecuteRemote.psm1  
  - ADMELauncherWinRM.psm1  
  - ADMELauncherWMI.psm1 |
| Get-WSManInstance | Retrieves an instance of a management resource that is specified by a resource Uniform Resource Identifier (URI). The information that is retrieved can be a complex XML information set, which is an object, or a simple value.  
  **MID Server script**  
  ScriptAPI.psm1 |
<table>
<thead>
<tr>
<th>Cmdlet/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group-Object</td>
<td>Displays objects in groups based on the value of a specified property. Group-Object returns a table with one row for each property value and a column that displays the number of items with that value. <strong>MID Server script</strong> RegistryAPI.psm1</td>
</tr>
<tr>
<td>Import-Module</td>
<td>Adds one or more modules to the current session. The modules that you import must be installed on the local computer or a remote computer. <strong>Probes</strong> Windows - Storage 2012 - PS (storage.2012.ps1) <strong>MID Server script</strong> Credentials.psm1 PSScript.ps1</td>
</tr>
<tr>
<td>Invoke-Command</td>
<td>Runs commands on a local or remote computer and returns all output from the commands, including errors. By using a single Invoke-Command command, you can run commands on multiple computers. <strong>Probes</strong> Windows - Storage 2012 - PS (storage.2012.ps1) <strong>MID Server script</strong> ADMELauncherWinRM.psm1 Credentials.psm1 ExecuteRemote.psm1 LaunchProc.psm1 Get-PEB.psm1 WMIFetch.psm1</td>
</tr>
<tr>
<td>Invoke-Expression</td>
<td>Evaluates or runs a specified string as a command and returns the results of the expression or command. Without Invoke-Expression, a string submitted at the command line would be returned (echoed) unchanged. <strong>MID Server script</strong> NmapInstallation.psm1 ScriptAPI.psm1</td>
</tr>
<tr>
<td>Invoke-WmiMethod</td>
<td>Calls the methods of Windows Management Instrumentation (WMI) objects. <strong>MID Server script</strong> LaunchProc.psm1 ADMELauncherWinRM.psm1 ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Measure-Object</td>
<td>Calculates the property values of certain types of object. Measure-Object performs three types of measurements, depending on the parameters in the command.</td>
</tr>
<tr>
<td>· Sum</td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td>New-Item</td>
<td>Creates a new item and sets its value. The types of items that can be created depend on the location of the item. For example, in the file system, New-Item creates files and folders. In the registry, New-Item creates registry keys and entries.</td>
</tr>
<tr>
<td>· ItemType</td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td>· Path</td>
<td>CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td>· Force</td>
<td>ExecuteRemote.psm1</td>
</tr>
<tr>
<td>· ErrorAction</td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td></td>
<td>NmapInstallation.psm1</td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWinRM.psm1</td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td>New-Object</td>
<td>Creates an instance of a Microsoft .NET Framework or COM object.</td>
</tr>
<tr>
<td>· TypeName</td>
<td><strong>Probes</strong></td>
</tr>
<tr>
<td>· Property</td>
<td>Windows - AWS Relationship (AWS_InstanceID.ps1)</td>
</tr>
<tr>
<td>· ArgumentList</td>
<td>Windows - Azure (is.azure.ps1)</td>
</tr>
<tr>
<td></td>
<td>Windows - MSSQL (sqlinfo.ps1)</td>
</tr>
<tr>
<td></td>
<td>Windows - File Discovery (filebaseddiscovery.ps1)</td>
</tr>
<tr>
<td></td>
<td>Windows - Storage 2012 - PS (storage.2012.ps1)</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td></td>
<td>CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td></td>
<td>Credentials.psm1</td>
</tr>
<tr>
<td></td>
<td>Get-PEB.psm1</td>
</tr>
<tr>
<td></td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td></td>
<td>WMIFetch.psm1</td>
</tr>
<tr>
<td></td>
<td>XMLUtil.psm1</td>
</tr>
<tr>
<td></td>
<td>ExecuteRemote.psm1</td>
</tr>
<tr>
<td>New-PSDrive</td>
<td>Creates temporary and persistent drives that are mapped to or associated with a location in a data store, such as a network drive, a directory on the local computer, or a registry key, and persistent Windows mapped network drives that are associated with a file system location on a remote computer.</td>
</tr>
<tr>
<td>· PSPProvider</td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td>· Root</td>
<td>ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td>· ErrorAction</td>
<td>ScriptAPI.psm1</td>
</tr>
<tr>
<td>· Credential</td>
<td>Name</td>
</tr>
<tr>
<td>· Scope</td>
<td></td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| **New-Variable**  | Creates a new variable in Windows PowerShell. You can assign a value to the variable while creating it or assign or change the value after it is created.  
**MID Server script**  
PSScript.ps1 |
| **New-PSSession** | Creates a Windows PowerShell session (PSSession) on a local or remote computer. When you create a PSSession, Windows PowerShell establishes a persistent connection to the remote computer.  
**MID Server script**  
PSRemoteSession.psm1 |
| **New-PSSessionOption** | Creates an object that contains advanced options for a user-managed session. You can use the object as the value of the SessionOption parameter of cmdlets that create a PSSession, such as a New-PSSession, Enter-PSSession, and Invoke-Command.  
**MID Server script**  
PSRemoteSession.psm1 |
| **New-Timespan**  | Creates a timespan object which represents an interval. You can use to add or subtract time from a DateTime object.  
**Probes**  
Windows - File Discovery (filebaseddiscovery.ps1) |
| **Out-File**      | Sends output to a file. You can use this instead of > operator when you need to use its parameters.  
**Probes**  
Windows - File Discovery (filebaseddiscovery.ps1) |
| **Out-Null**      | Sends output to NULL, in effect, deleting it.  
**MID Server script**  
Get-PEB.psm1 |
| **Out-String**    | Converts the objects that Windows PowerShell manages into an array of strings. By default, Out-String accumulates the strings and returns them as a single string, but you can use the stream parameter to direct Out-String to return one string at a time. This cmdlet lets you search and manipulate string output as you would in traditional shells when object manipulation is less convenient.  
**MID Server script**  
Get-PEB.psm1 |
<table>
<thead>
<tr>
<th>Cmdlet/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receive-Job</td>
<td>Gets the results of Windows PowerShell background jobs, such as those started by using the Start-Job cmdlet or the AsJob parameter of any cmdlet. You can get the results of all jobs or identify jobs by their name, ID, instance ID, computer name, location, or session, or by submitting a job object.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td>Remove-Item</td>
<td>Deletes one or more items. Because it is supported by many providers, it can delete many different types of items, including files, folders, registry keys, variables, aliases, and functions.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>ExecuteRemote.psm1</td>
</tr>
<tr>
<td></td>
<td>NmapInstallation.psm1</td>
</tr>
<tr>
<td></td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td></td>
<td>CollectConnectionsAndProcessesInfo.psm1</td>
</tr>
<tr>
<td>Remove-Job</td>
<td>Deletes Windows PowerShell background jobs. You can start jobs by using the Start-Job or the AsJob parameter of any cmdlet. You can use this cmdlet to delete all jobs or delete jobs based on their name, ID, instance ID, command, or state, or by passing a job object to Remove-Job. Without parameters or parameter values, Remove-Job has no effect.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td>Remove-PSDrive</td>
<td>Deletes temporary Windows PowerShell drives that were created by using the New-PSDrive cmdlet.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>ADMELauncherWMI.psm1</td>
</tr>
<tr>
<td></td>
<td>ScriptAPI.psm1</td>
</tr>
<tr>
<td>Remove-PSSession</td>
<td>Closes Windows PowerShell sessions (PSSessions) in the current session. It stops any commands that are running in the PSSessions, ends the PSSession, and releases the resources that the PSSession was using. If the PSSession is connected to a remote computer, this cmdlet also closes the connection between the local and remote computers.</td>
</tr>
<tr>
<td></td>
<td><strong>Probes</strong></td>
</tr>
<tr>
<td></td>
<td>Windows - Storage 2012 - PS (storage.2012.ps1)</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>Credentials.psm1</td>
</tr>
<tr>
<td>Remove-Variable</td>
<td>Deletes a variable and its value from the scope in which it is defined, such as the current session. You cannot use this cmdlet to delete variables that are set as constants or those that are owned by the system.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>PSScript.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Rename-Item</td>
<td>Changes the name of a specified item. This cmdlet does not affect the content of the item being renamed. You cannot use Rename-Item to move an item, such as by specifying a path together with the new name. To move and rename an item, use the Move-Item cmdlet.</td>
</tr>
<tr>
<td></td>
<td><strong>Probes</strong></td>
</tr>
<tr>
<td></td>
<td>Windows - File Discovery (filebaseddiscovery.ps1)</td>
</tr>
<tr>
<td>Select-Object</td>
<td>Selects specified properties of an object or set of objects. It can also select unique objects, a specified number of objects, or objects in a specified position in an array. Alias is &quot;Select&quot;.</td>
</tr>
<tr>
<td></td>
<td><strong>Probes</strong></td>
</tr>
<tr>
<td></td>
<td>Windows - MSSQL (sqlinfo.ps1)</td>
</tr>
<tr>
<td></td>
<td>MID Server script</td>
</tr>
<tr>
<td></td>
<td>- CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td></td>
<td>- RegistryAPI.psm1</td>
</tr>
<tr>
<td>Select-String</td>
<td>Searches for text and text patterns in input strings and files. You can use it like Grep in UNIX and Findstr in Windows. You can type Select-String or its alias, sls.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td>Select-XML</td>
<td>Allows you to use XPath queries to search for text in XML strings and documents. Enter an XPath query, and use the Content, Path, or Xml parameter to specify the XML to be searched.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>ScriptApi.psm1</td>
</tr>
<tr>
<td>Set-Content</td>
<td>Writes or replaces the content in the specified item, such as a file. You can type the content in the command or send content through the pipeline to Set-Content.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>- ADMELauncherWinRM.psm1</td>
</tr>
<tr>
<td></td>
<td>- CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td></td>
<td>- ExecuteRemote.psm1</td>
</tr>
<tr>
<td>Set-Location</td>
<td>Sets the working location to a specified location. That location could be a directory, a sub-directory, a registry location, or any provider path.</td>
</tr>
<tr>
<td></td>
<td><strong>MID Server script</strong></td>
</tr>
<tr>
<td></td>
<td>Credentials.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>Set-StrictMode</strong></td>
<td>Configures strict mode for the current scope and all child scopes, and turns it on and off. When strict mode is on, Windows PowerShell generates a terminating error when the content of an expression, script, or script block violates basic best-practice coding rules.</td>
</tr>
<tr>
<td><strong>Set-Variable</strong></td>
<td>Assigns a value to a specified variable or changes the current value. If the variable does not exist, the cmdlet creates it.</td>
</tr>
<tr>
<td><strong>Sort-Object</strong></td>
<td>Sorts objects in ascending or descending order based on the values of properties of the object. You can specify a single property or multiple properties (for a multi-key sort), and you can select a case-sensitive or case-insensitive sort. You can also direct Sort-Object to display only the objects with a unique value for a particular property.</td>
</tr>
<tr>
<td><strong>Split-Path</strong></td>
<td>Returns only the specified part of a path, such as the parent folder, a subfolder, or a file name. It can also get items that are referenced by the split path and tell whether the path is relative or absolute.</td>
</tr>
</tbody>
</table>

### MID Server script
- Set-PEB.psm1
- CollectConnectionsAndProcessesInfo.psm1
- ExecuteRemote.psm1

### MID Server script
- PSScript.psm1
- DiagnosticsUtil.psm1

### Probes
- Windows - MSSQL (sqlinfo.ps1)
- Windows - Weblogic - Find (findconfig.ps1)
- Windows - Storage 2012 - PS (storage.2012.ps1)

### MID Server script
- Credentials.psm1
- LaunchProc.psm1
- PSScript.psm1
- ExecuteRemote.psm1
<table>
<thead>
<tr>
<th>Cmdlet/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Start-Sleep</strong></td>
<td>Suspends the activity in a script or session for the specified period of time. You can use it while waiting for an operation to complete or pausing before repeating an operation.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td>Windows - Active Connections Multiple (WMI_ActiveConnections.ps1)</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>List of probe scripts</td>
</tr>
<tr>
<td><strong>Stop-Job</strong></td>
<td>Stops Windows PowerShell background jobs that are in progress. You can use this cmdlet to stop all jobs or stop selected jobs based on their name, ID, instance ID, or state, or by passing a job object to Stop-Job.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>LaunchProc.psm1</td>
</tr>
<tr>
<td><strong>Test-Path</strong></td>
<td>Determines whether all elements of the path exist. It returns $True if all elements exist and $False if any are missing. It can also tell whether the path syntax is valid and whether the path leads to a container or a terminal or leaf element.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>ExecuteRemote.psm1, LaunchProc.psm1, NmapInstallation.ps1, PSScript.ps1, RegistryAPI.psm1, ADMELauncherWinRM.psm1, ADMELauncherWMI.psm1, CollectConnectionsAndProcessesInfo.ps1</td>
</tr>
<tr>
<td><strong>Test-WSMan</strong></td>
<td>Submits an identification request that determines whether the WinRM service is running on a local or remote computer. If the tested computer is running the service, the cmdlet displays the WS-Management identity schema, the protocol version, the product vendor, and the product version of the tested service.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>ScriptAPI.psm1</td>
</tr>
<tr>
<td><strong>Where-Object</strong></td>
<td>Selects objects that have particular property values from the collection of objects that are passed to it. For example, you can use the Where-Object cmdlet to select files that were created after a certain date, events with a particular ID, or computers that use a particular version of Windows.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td>CollectConnectionsAndProcessesInfo.ps1, DiagnosticsUtil.psm1, WMIFetch.psm1</td>
</tr>
<tr>
<td>Cmdlet/Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Write-Error</td>
<td>Declares a non-terminating error. By default, errors are sent in the error stream to the host program to be displayed, along with output.</td>
</tr>
<tr>
<td>· Message</td>
<td></td>
</tr>
<tr>
<td>· Category</td>
<td></td>
</tr>
<tr>
<td>Probes</td>
<td></td>
</tr>
<tr>
<td>· Windows - MySQL Configuration (mysqlconfig.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - MySQL Version (mysqlinfo.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Oracle - Instance Pfile (oraclepfile.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Oracle - Instance Version (oracleversion.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - MSSQL (sqlinfo.ps1)</td>
<td></td>
</tr>
<tr>
<td>MID Server script</td>
<td></td>
</tr>
<tr>
<td>· ADMELauncherWinRM.psm1</td>
<td></td>
</tr>
<tr>
<td>· ADMELauncherWMI.psm1</td>
<td></td>
</tr>
<tr>
<td>· ExecuteRemote.psm1</td>
<td></td>
</tr>
<tr>
<td>· LaunchProc.psm1</td>
<td></td>
</tr>
<tr>
<td>· NmapInstallation.ps1</td>
<td></td>
</tr>
<tr>
<td>Write-Host</td>
<td>Writes customized output to a host. You can specify the color of text by using the ForegroundColor parameter, and you can specify the background color by using the BackgroundColor parameter. The Separator parameter lets you specify a string to use to separate displayed objects. The particular result depends on the program that is hosting Windows PowerShell.</td>
</tr>
<tr>
<td>· NoNewLine</td>
<td></td>
</tr>
<tr>
<td>Probes</td>
<td></td>
</tr>
<tr>
<td>· Windows - Network ARP Table (arpTable.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - Weblogic - Find (findconfig.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - JBoss Get (findjbossconfigxml.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - WebSphere - Web Services ()</td>
<td></td>
</tr>
<tr>
<td>· Windows - JBoss Find (findwebxml.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - Weblogic - Get ()</td>
<td></td>
</tr>
<tr>
<td>· Windows - WebSphere - Cell (getcellxml.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - WebSphere - Web Applications (getserverindexxml.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Oracle - Instance Pfile (oraclepfile.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Oracle - Instance Version (oracleversion.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - MSSQL (sqlinfo.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - Storage 2012 - PS (storage.2012.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - Storage 2008 - PS (storage.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - Active Connections Multiple (WMI_ActiveConnections.ps1)</td>
<td></td>
</tr>
<tr>
<td>MID Server script</td>
<td></td>
</tr>
<tr>
<td>· ADMELauncherWinRM.psm1</td>
<td></td>
</tr>
<tr>
<td>· ADMELauncherWMI.psm1</td>
<td></td>
</tr>
<tr>
<td>· PSScript.ps1</td>
<td></td>
</tr>
<tr>
<td>· RegistryAPI.psm1</td>
<td></td>
</tr>
<tr>
<td>· ScriptAPI.psm1</td>
<td></td>
</tr>
<tr>
<td>· WMIFetch.psm1</td>
<td></td>
</tr>
<tr>
<td>· DiagnosticsUtil.psm1</td>
<td></td>
</tr>
<tr>
<td>· NmapInstallation.ps1</td>
<td></td>
</tr>
</tbody>
</table>
### Cmdlet/Parameters

<table>
<thead>
<tr>
<th>Cmdlet/Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write-Output</td>
<td>Sends the specified objects to the next command in the pipeline. If the command is the last command in the pipeline, the objects are displayed in the console.</td>
</tr>
<tr>
<td><strong>Probes</strong></td>
<td></td>
</tr>
<tr>
<td>· Windows - Azure (is.azure.ps1)</td>
<td></td>
</tr>
<tr>
<td>· ExecuteRemote (testExecuteRemote.ps1)</td>
<td></td>
</tr>
<tr>
<td>· Windows - File Discovery (filebaseddiscovery.ps1)</td>
<td></td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td></td>
</tr>
<tr>
<td>Get-PEB.psm1</td>
<td></td>
</tr>
<tr>
<td>Write-Verbose</td>
<td>Writes text to the verbose message stream in Windows PowerShell. Typically, the verbose message stream is used to deliver information about command processing that is used for debugging a command.</td>
</tr>
<tr>
<td><strong>MID Server script</strong></td>
<td></td>
</tr>
<tr>
<td>· Get-PEB.psm1</td>
<td></td>
</tr>
<tr>
<td>· LaunchProc.psm1</td>
<td></td>
</tr>
<tr>
<td>· ExecuteRemote.psm1</td>
<td></td>
</tr>
</tbody>
</table>

### MID Server parameters for PowerShell

Parameters control the behavior of a particular MID Server and have lower precedence than MID Server properties.

The following parameters are optional.

**Note:** After changing the setting for any parameter, be sure to restart the MID Server service.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid.monitoring.force_sigar_cpu_monitoring</td>
<td>Enables CPU monitoring on Windows MID Server hosts using Java 11. An outdated SIGAR library causes the MID Server to crash. If you patch the SIGAR library, set this value to True to enable CPU Monitoring for Windows with Java 11.</td>
</tr>
<tr>
<td>· Type: True</td>
<td>False</td>
</tr>
<tr>
<td>· Default value: false</td>
<td></td>
</tr>
<tr>
<td>mid.powershell_api.session_pool.max_size</td>
<td>Specifies the maximum number of sessions allowed in the session pool.</td>
</tr>
<tr>
<td><strong>Note:</strong> Setting or changing this parameter requires restarting the MID Server.</td>
<td></td>
</tr>
<tr>
<td>· Type: Integer</td>
<td></td>
</tr>
<tr>
<td>· Default value: 25</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>mid.powershell_api.session_pool.target.max_size</td>
<td>Specifies the maximum number of sessions allowed in the pool per target host.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Setting or changing this parameter requires restarting the MID Server.</td>
</tr>
<tr>
<td></td>
<td>• Type: Integer</td>
</tr>
<tr>
<td></td>
<td>• Default value: 2</td>
</tr>
<tr>
<td>mid.powershell_api.winrm.use_ssl</td>
<td>Requires the use of SSL certificates for HTTPS connections using WinRM.</td>
</tr>
<tr>
<td></td>
<td>• Type: True</td>
</tr>
<tr>
<td></td>
<td>• Default value: false</td>
</tr>
<tr>
<td>mid.powershell_api.winrm.remote_https_port</td>
<td>Configures the port for connecting to Windows servers using PowerShell over HTTPS.</td>
</tr>
<tr>
<td></td>
<td>• Type: Integer</td>
</tr>
<tr>
<td></td>
<td>• Default value: 5986</td>
</tr>
<tr>
<td>mid.powershell_api.winrm.remote_port</td>
<td>Configures the port for connecting to Windows servers using PowerShell over HTTP.</td>
</tr>
<tr>
<td></td>
<td>• Type: Integer</td>
</tr>
<tr>
<td></td>
<td>• Default value: 5985</td>
</tr>
<tr>
<td>mid.powershell_api.winrm.skip_ssl_cert_check</td>
<td>Skips the SSL certificate check when using WinRM for HTTPS connections.</td>
</tr>
<tr>
<td></td>
<td>• Type: True</td>
</tr>
<tr>
<td></td>
<td>• Default value: false</td>
</tr>
<tr>
<td>mid.powershell_api.winrm.skip_ssl_cert_check_options</td>
<td>Skips specific SSL certificate checks when using WinRM for HTTPS connections. Configure the MID Server to skip checks for certificates from a Certification Authority (CA), from the Common Name (CN) that identifies the host associated with the certificate, and for revoked certificates.</td>
</tr>
<tr>
<td></td>
<td>• Type: String</td>
</tr>
<tr>
<td></td>
<td>• Default value: -SkipCACheck -SkipCNCheck -SkipRevocationCheck</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>mid.powershell.use_credentials</td>
<td>Determines the credentials to use for Discovery with PowerShell. A setting of true directs the MID Server to run probes with the Windows credentials from the credentials table. To run probes with the credentials of the user for the MID Server service, set this parameter to false.</td>
</tr>
<tr>
<td></td>
<td>• Type: true</td>
</tr>
<tr>
<td></td>
<td>• Default value: true</td>
</tr>
<tr>
<td>mid.use_powershell</td>
<td>Enables or disables PowerShell for Discovery. Restart the MID Server after changing the value. If PowerShell is not installed or the version installed is less than version 2.0, Discovery reverts to using WMIRunner.</td>
</tr>
<tr>
<td></td>
<td>• Type: true</td>
</tr>
<tr>
<td></td>
<td>• Default value: true, in the Fuji release.</td>
</tr>
<tr>
<td></td>
<td>• Default value: false, in releases prior to Fuji.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>mid.powershell.path</td>
<td>Enables an administrator to point to a specific PowerShell on a MID Server in cases where more than one PowerShell is installed. Supply the path to the folder containing the PowerShell executable, for example, C:\mypowershell or C:\mypowershell\ServiceNow automatically appends the string powershell.exe to the path. Configure this parameter when both a 32-bit and 64-bit PowerShells are active on the same MID Server, and it becomes necessary to launch the correct PowerShell for the context. 64-bit Windows employs file system redirection and the MID Server runs as a 32-bit application. If trying to specify a path in %WinDir%\System32, Windows automatically redirects to %WinDir%\SysWOW64. To avoid redirection, specify the path as %WinDir%\Sysnative. For example, instead of C:\WINDOWS\system32\WindowsPowerShell\v1.0, specify C:\WINDOWS\sysnative\WindowsPowerShell\v1.0. Note: On a 64-bit version of Windows XP, a Microsoft hotfix may be required to enable this. To discover applications running on a 64-bit Windows machine, the MID Server must be running on a 64-bit Windows host machine.</td>
</tr>
<tr>
<td>mid.powershell.enforce_utf8</td>
<td>Enable this parameter to force commands on a target Windows system to return UTF-8 encoded output. Disabling it allows the target system to use its default encoding. This parameter is only valid when PowerShell is enabled. Setting this value to false may result in incorrect values in the CMDB when non-ASCII characters are returned by a probe.</td>
</tr>
<tr>
<td></td>
<td>· Type: string (path)</td>
</tr>
<tr>
<td></td>
<td>· Default value: none</td>
</tr>
<tr>
<td></td>
<td>· Type: true</td>
</tr>
<tr>
<td></td>
<td>· Default value: true</td>
</tr>
</tbody>
</table>
### Application Dependency Mapping (ADM) for Discovery

Application Dependency Mapping (ADM) discovers CIs by detecting TCP connections between devices.
When ADM discovery is enabled, Discovery always runs the ADM probes during the exploration stage to find the TCP traffic on your network. Using these TCP connections, Discovery can find additional CIs and create relationships between them.

**Standard and enhanced ADM discovery**

**Standard ADM discovery**

Triggers an ADM probe to take a single snapshot of TCP connections on target machines using `netstat` and `lsof` commands. However, it does not detect new TCP connections that occur between these 24-hour intervals.

Standard ADM discovery is enabled by default.

**Enhanced ADM discovery**

Triggers an enhanced ADM probe that installs a script on host machines on which Discovery takes regular samples of active TCP connections. You can use a set of properties and probe parameters that control several aspects of enhanced ADM discovery.

Enhanced ADM discovery is not enabled by default.

Service Mapping uses the ADM probes for traffic-based discovery. See [Traffic-based discovery in Service Mapping](#) for more information.

**Requirements**

Both standard and enhanced ADM discovery support the following operating systems:

- Windows
- Unix
- Solaris (On Solaris machines, enhanced ADM is available in global zones. Only standard ADM is available for local zones.)
- AIX
- HP-UX

To use the WMI protocol, **version 3.0 of PowerShell** is required.

**TCP tracking file for enhanced ADM**

The first time that Discovery launches the enhanced probe, no TCP information is gathered. Rather, the enhanced probe only deploys the TCP tracking file, in which information about detected TCP connections is saved. After Discovery runs again, the enhanced probe relays the contents of the TCP tracking file back to the instance via the MID Server. The information is saved in the CMDB.

The TCP tracking file is saved in these locations by default:

- **Windows**: `admin$\temp\{your_instance_name}`. While TCP connections are being monitored, the information is stored in memory and then later saved to the file.
- **Linux**: `/tmp/{your_instance_name}`.

You can change the location and other aspects of the tracking file with ADM properties. If you disable enhanced ADM, Discovery cleans up the folders from the host file system the next time Discovery runs, but the TCP tracking file remains.
Requirements for Enhanced ADM on Linux machines

Modify the sudoers file to include these commands:

**Windows**
- Utility tool support: netstat
- Required sudoer commands: None
- Additional requirements: None

**AIX**
- Utility tool support:
  - netstat
  - lsof (if netstat does not work)
- Required sudoer commands:
  - `Cmd_Alias ADME_CMDS=/usr/bin/netstat -Aan, /usr/sbin/lsof -iTCP -n -P`
  - `discoUser ALL=(root) NOPASSWD:ADME_CMDS`
  - `Defaults:discoUser ! requiretty`
- Additional requirements: None

**Solaris**
- Utility tool support:
  - netstat (for Solaris version 11.2)
  - lsof (for Solaris versions prior to 11.2)
- Required sudoer commands:
  - `Cmd_Alias ADME_CMDS = /usr/bin/netstat,/opt/csw/bin/lsof -iTCP -n -P -zglobal`
  - `discoUser ALL=(root) NOPASSWD:ADME_CMDS`
  - `Defaults!ADME_CMDS !requiretty`
  - `Cmd_Alias ADME_PS = /usr/ucb/ps`
  - `discoUser ALL=(root) NOPASSWD:ADME_PS`
  - `Defaults!ADME_PS !requiretty`

- **Note:** You must add `-zglobal` for lsof to run.

- Additional requirements: Add the lsof path to the secure path. For example, if the path to lsof is `/opt/csw/bin`, the secure path should be `secure_path = /usr/bin:/usr/sbin:/bin:/sbin:/opt/csw/bin`

- **Note:** The commands with `PS` apply to standard ADM as well as enhanced ADM.

**HP UX**
- Utility tool support: lsof
- Required sudoer commands:
  - `Cmd_Alias ADME_CMDS = /usr/local/bin/lsof -iTCP -n -P`
**What to do (enhanced ADM)**

To set up and use enhanced ADM, perform these steps:

1. *Enable enhanced ADM with the glide.discovery.enable_adme Discovery property.*
2. Determine the base directory where the TCP tracking file is placed on the host computers. If you want to change the location, *specify it in these Discovery properties:*
   - glide.discovery.adme.base_dir_unix
   - glide.discovery.adme.base_dir_windows
3. Do either of the following based on the operating systems of the host computers:
   - Linux: *modify the sudoers file* to include the necessary commands.
   - Windows: Set the mid.windows.management_protocol parameter on the MID Server to use WinRM or WMI. This parameter sets the protocol that your MID Server uses to communicate with remote Windows hosts.
4. *Create a schedule and run Discovery.*
5. *View the results in the TCP tables.*
6. Optionally control both the size of the TCP tracking file and the payload using parameters on the enhanced ADM probes. See *TCP connection discovery* for the names of the probes and parameters.

**Configure ADM discovery parameters and properties**

Using properties, you can enable or disable ADM discovery, and you can set the location of the TCP tracking file for enhanced ADM.

Role required: admin

---

**Centos and Ubuntu**

- **Utility tool support:** None
- **Required sudoer commands:**
  - Cmnd_Alias ADME_CMDS = /usr/sbin/lsof -iTCP -n -P, /usr/bin/netstat -antp
  - discoUser ALL=(root) NOPASSWD:ADME_CMDS
  - Defaults!ADME_CMDS !requiretty
- **Additional requirements:** None

Note the following for the sudo commands:

- ADME_CMDS is a variable referring to a list of commands.
- /usr/bin/ is the directory where your lsof and netstat resides.
- discoUser is a user that must have root access.

---

**Note:** Netstat is not supported.
You can also control both the size of the TCP tracking file and the payload using parameters on
the enhanced ADM probes. See TCP connection discovery for the names of the probes and
parameters.

1. Navigate to System Properties > Discovery.
2. Configure the following properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Type</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.enable_adme</td>
<td>ADME: Enable enhanced ADM probe. If “yes”, the ADM Enhanced probe will be triggered and only fall back to the ADM probe as needed.</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>glide.discovery.application_mapping</td>
<td>Enable the application mapping feature. When this property is set to true, all discoveries trigger the relevant ADM probe during the exploration phase if Discovery finds a server.</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>glide.discovery.adme.sampling_interval</td>
<td>ADME - Sampling interval (sec): How often to sample process and connection data in seconds. Must be a minimum of 5 seconds.</td>
<td>integer</td>
<td>120</td>
</tr>
<tr>
<td>glide.discovery.adme.base_dir_unix</td>
<td>ADME: Unix Base Dir: An existing directory on the target Unix machines to be used as a workspace. Must be an absolute path to the directory. The UNIX and Linux credentials that you use to connect to the device must also be able to access this folder.</td>
<td>string</td>
<td>/tmp</td>
</tr>
<tr>
<td>glide.discovery.adme.base_dir_windows</td>
<td>ADME: Windows Base Dir: A network share on the target Windows machines to be used as a workspace. If you change the directory, it must also be a network shared folder.</td>
<td>string</td>
<td>admin$\temp</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| glide.discovery.adme.aggregation_interval | ADME - Aggregation interval (sec): How often to aggregate instantaneous data into chunks in seconds. It must be a minimum of 60 seconds and it must be a multiple of ADME Sampling interval.  
  - **Type**: integer  
  - **Default value**: 3600 |
| glide.discovery.adme.rolling_window_size | ADME - Rolling window size: How many recently aggregated chunks to keep. An aggregated chunk consists of ADME Aggregation interval divided by ADME Sampling interval samples. Only the newest chunks will be kept.  
  - **Type**: integer  
  - **Default value**: 24 |
| glide.discovery.adme.max_total_samples | ADME - Max total samples: The limit of samples being taken by the script. The script will stop running after this amount of samples are taken.  
  - **Type**: integer  
  - **Default value**: 100 |

3. **Click Save.**

**Discovery behaviors**

Discovery behaviors determine the probes that Shazzam launches, and from which MID Servers these probes are launched.

Unlike a scan performed by a single MID Server on a designated IP address range, a behavior can assign different tasks to multiple MID Servers on the same IP address segment or on different network segments. Behaviors are available in **Discovery schedules** for discoveries in which configuration items (CI) are updated in the CMDB.

Behaviors can be used in the following scenarios:

- **Load balancing**: A behavior enables load balancing in systems that use multiple MID Servers deployed across one or more domains.
- **Multiple protocols in multiple domains**: Configure one MID Server to scan for all protocols on one domain and another MID Server to perform a WMI scan on a second domain.
- **Access Control Lists (ACL)**: Discovery can scan SNMP devices protected by an ACL if the MID Server host machine is granted access by that ACL. Use a behavior to configure a MID Server to scan devices protected by an ACL.
- **Devices running two protocols**: Some devices might have two protocols running at the same time. Examples of this are the SSH and SNMP protocols running concurrently on one device (most common). A behavior can control which of the two protocols is explored for certain devices. The behavior then prevents the other protocol from being explored.
Behaviors and domain separation

Behaviors also enable the efficient Discovery of SSH and SNMP devices and WMI devices running on multiple Windows domains, using multiple MID Servers.

For example, an organization has two Windows domains in its network and a variety of UNIX computers and SNMP devices. The challenge is to discover all the devices efficiently, without duplicating effort. Each domain contains a Windows MID Server which is used to scan the IP addresses from the two domains specified in the Discovery Schedule, as well as the SSH and SNMP devices. We need a Behavior that divides the work appropriately to avoid scanning anything twice. In this example, we assume that both domains are in the same geographical location, and that a single schedule is sufficient.

Note: The preferred method for running Discovery over multiple Windows domains is to use PowerShell, which allows a single MID Server to authenticate on machines on different domains using credentials stored on the instance.

Create a Discovery behavior

Create a Discovery behavior to determine which probes Shazzam launches and which MID Server is used.

Role required: discovery_admin

1. Navigate to Discovery Definition > Behavior and click New.
2. Enter a name.
3. Right-click the form header and select Save.
4. In the Discovery Functionality related list, click **New**.
   Discovery Functionality defines what each MID Server in this behavior must do, specifically which protocols to detect.

5. Fill out the form fields:
### Field | Description
--- | ---
**Phase** | Enter an integer that represents an arbitrary phase. The phase is used to group one or more functionalities together. All the functionalities within a specified phase are executed together, and all phases are executed in numerical order. All functionalities in a behavior can have the same phase.

The Shazzam probe runs once for each phase in a behavior, which makes fewer phases desirable. Run multiple phases for behaviors only when devices in the network are running multiple protocols, such as SSH and SNMP. In that example, set one phase for the SSH scan and another phase for the SNMP scan.

**Active** | Keep this option selected to apply the discovery functionality.

**Functionality definition** | Click the lookup icon, and then select a pre-configured functionality that defines the protocol or list of protocols that each MID Server scans.

**Match criteria** | Define criteria here for Windows MID Servers.

**MID Servers** | Select one or more MID Servers to perform this functionality for the following Discovery types:
- IP Scan
- CI Scan

Discovery automatically balances the load when multiple MID Servers are selected.

---

6. Right-click the form header and select **Save**.

7. To add criteria that the functionality must meet in order to be triggered, click **New** in the Functionality Criteria related list, fill out the form fields, and then click **Submit**.

### Field | Input Value
--- | ---
**Name** | The name in the criteria is the variable that passes the following information:
- `mid_server`: MID Server that processes the results from the Shazzam probe.
- `win_domain`: Windows domain of the target device.

**Operator** | Select a logical operator.
Enter the actual name of the MID Server (\texttt{mid	extunderscore server}) or domain (\texttt{win	extunderscore domain}) to pass to Discovery for this criteria. This field can also have a value of \texttt{mid	extunderscore domain}, which defines the Windows domain of the MID Server that is processing the Shazzam results.

\textbf{Note:} Functionality criteria are required for Windows MID Servers only, and only when the behavior controls Discovery across multiple domains. When the instance launches the Shazzam probe for a Discovery in which a behavior defines multiple MID Servers to scan multiple domains, the functionality criteria determine which MID Server process the results of the probe.

The following graphic shows an example of functionality criteria.

\textit{Create a Discovery Schedule} of type \textbf{Configuration Item}, and select \textbf{Use Behavior} for the MID Server selection method.

\textit{Set up a load balancing behavior}

When multiple MID Servers are configured to scan the same protocol, users can set up a load balancing behavior to automatically balance the work between the MID Servers.

Behaviors enable load balancing when multiple MID Servers are configured to scan the same protocol. For example, an organization has fifty remote locations, varying in size from 10 devices to several hundred devices. All the satellite offices are part of the same Windows domain, which is administered from a central location. There are three MID Servers installed at this central location: two to scan all the Windows devices and one to scan the remaining devices. Because some of the remote offices are in different time zones, different schedules must be created to run Discovery at off-peak hours in each time zone. The same behavior can be used for each schedule, and the behavior will load-balance the two Windows MID Servers automatically.

1. Create the behavior record.
   a) Create a new Discovery behavior record, and name it something descriptive, such as \texttt{LoadBalanced}.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
b) Right-click in the header bar and select **Save** from the pop-up menu.

   This action saves the record and creates the Discovery Functionality related list in the form.

2. Define the Windows functionality.
   a) Click **New** in the Discovery Functionality related list.
   b) Create a new record using the fields described here.
   c) Right-click in the header bar and select **Save** from the pop-up menu.

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select Windows only (WMI) from the list. This functionality defines the protocol that will be scanned. Because we selected to scan WMI, we must select Windows MID Servers for this functionality.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>We select the two MID Servers that we want to share the load for the WMI probes. By entering multiple MID Servers in this field, we tell Discovery to balance the load between these servers automatically. If we were to create separate functionality for each server, load balancing would not occur.</td>
</tr>
<tr>
<td>Active</td>
<td>Make sure this check box is selected to enable this behavior.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Leave the default criteria of Any.</td>
</tr>
</tbody>
</table>
3. Create Functionality Criteria.

**Note:** All Windows functionality requires criteria to identify the Windows domain and define any special behavior for the MID Servers named.

a) Click **New** in the related list.

b) Enter the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>We enter <code>win_domain</code> to name the Windows domain that Discovery will scan with the MID Servers we have defined.</td>
</tr>
<tr>
<td>Field</td>
<td>Input Value</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Operator</td>
<td>Select equals as the operator in this criteria.</td>
</tr>
<tr>
<td>Value</td>
<td>This is the name of the Windows domain that these MID Servers will scan for devices.</td>
</tr>
<tr>
<td>Active</td>
<td>Be sure to enable the criteria by selecting this check box (true).</td>
</tr>
</tbody>
</table>

The completed criteria appear in the Discovery Functionality form for the Windows MID Servers.
4. Define the functionality for the remaining scans.

Create a second record to scan for all the remaining protocols (SSH, SNMP, etc), using the following settings:
<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field. This phase will be executed at the same time as the WMI scans and with the same Shazzam probe. We use the same phase number for efficiency and because we know that none of the devices in the target IP ranges are running multiple protocols (e.g. SSH and SNMP). If any devices were running multiple protocols, we would want to specify a second or even a third phase to discover the correct protocol first for each device.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select All except Windows (no WMI) from the list. This functionality causes the MID Server to scan all remaining protocols after Discovery has run the WMI scans.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>Name the MID Server that will scan for all other devices. If we want to use automatic load balancing, we can add an additional MID Server to this field.</td>
</tr>
</tbody>
</table>

The completed Discovery Behavior form looks like this. It is not necessary to create Functionality Criteria for this MID Server.
5. Create a **Discovery schedule** for each time zone and name the behavior we just created.
   a) Navigate to **Discovery** > **Discovery Schedules** and click **New** in the record list.
   b) Select a **Discover** type of Configuration items.
      This action displays the Behavior field.
   c) Click the magnifier icon and select the behavior to use. In our example, we select **LoadBalanced**.
   d) Select the appropriate time to run Discovery for this location.
e) Click **Quick Ranges** and define the IP address ranges, networks, or lists to scan for this location.

f) Save the record.

g) Create additional schedules for each time zone or region in the network and select the same behavior.
### Discovery Schedule

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>My Disco Schedule</td>
</tr>
<tr>
<td>Discover</td>
<td>Configuration Items</td>
</tr>
<tr>
<td>NIS Server selection method</td>
<td>Auto-Select NIS Server</td>
</tr>
<tr>
<td>Max run time</td>
<td>Days: 0, Hours: 10, 00, 00</td>
</tr>
<tr>
<td>Run</td>
<td>Weekly</td>
</tr>
<tr>
<td>Day (run, day, week)</td>
<td>Monday</td>
</tr>
<tr>
<td>Time</td>
<td>Hours: 02, 00, 00</td>
</tr>
</tbody>
</table>

### Advanced

- Include alive: **on**
- Shazzam batch size: 1,000
- Log state changes: **on**
- Shazzam cluster support: **on**
- Use SNMP version: 1.3.6.1.2.

### Related Links

- Discovery IP Ranges
  - Schedule = My Disco Schedule
    - Type
    - Summary
    - Active
    - IP Network: 10.11.0.0/10
      - true
Examples of Discovery behavior functionalities
This example of a Discovery behavior requires three functionalities for the behavior.

We will create three functionalities for this Behavior: one MID Server to scan Domain A for Windows devices only; a second functionality for the same MID Server to scan for all SSH and SNMP devices; and a third functionality that names a second MID Server to scan Domain B for Windows devices. The rationale for this is as follows:

- A Windows MID Server can only discover Windows machines on the Windows domain to which it is joined. This is entirely due to the way Windows authentication works. For this reason, we need a WMI functionality for each domain.
- A Windows MID Server, provided with the correct credentials, can discover SSH and SNMP devices anywhere; however, we cannot combine WMI, SSH, and SNMP functionalities across Windows domains. This is because the functionality criteria for the WMI scans locks in the Discovery to one specific domain. For this reason, SSH and SNMP discoveries require a separate functionality.
- We want to scan each machine only once.

Functionality 1: WMI Scanning on Domain A

We configure a MID Server to scan for the WMI protocol on Domain A. WMI scans authenticate on Windows machines using the domain credentials of the Windows MID Server machine. Windows MID Servers cannot scan for the WMI protocol outside their own domains.

Create the first functionality using the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field. All functionalities in this example use the same phase number, which launches a single Shazzam probe for all the functionalities in that phase. A single phase, when feasible, is the most efficient use of the Shazzam probe.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select Windows, DNS, and WINS from the list. This functionality defines the WMI protocol that will be scanned and resolves the domain. Because we selected to scan for WMI, we must select a Windows MID Server for this functionality.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>We select a Windows MID Server from Domain A - in this case sandb01-358.</td>
</tr>
<tr>
<td>Active</td>
<td>Make sure this check box is selected to enable this behavior.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Change the criteria to All.</td>
</tr>
</tbody>
</table>

Create Functionality Criteria

All Windows functionality requires criteria to identify the domain and the MID Server. In our example, we will create two criteria for this functionality. To create Functionality Criteria, click New in the Related List and enter the following values for the MID Server doing the WMI scanning on Domain A:
Create the following criteria:

- Enter `mid_server` to name the MID Server that will execute the WMI scans on Domain A.
- Enter `win_domain` to name the Windows domain that Discovery will scan with the MID Server defined.

**Operator**

Select **equals** as the operator in this criteria.

**Value**

- For the `mid_server` value, enter the name of the MID Server that will scan Domain A for Windows devices.
- For the `win_domain` value, enter the name of Domain A that this MID Server will scan for Windows devices.

Be sure to enable the criteria by selecting this check box (true).

The completed criteria appear in the Discovery Functionality form for this behavior.

**Functionality 2: SSH and SNMP**

In our network, we want to scan for UNIX computers and netgear, but we don’t want to classify these devices twice. One of our MID Servers will be configured to classify SSH and SNMP using a different functionality than it does for WMI scans. We do not need to create criteria for non-WMI functionality.

Create the second functionality using the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field. All functionalities in this example use the same phase number, which launches a single Shazzam probe for all the functionalities in that phase. A single phase, when feasible, is the most efficient use of the Shazzam probe.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select <strong>All except Windows (no WMI)</strong> from the list. This functionality will scan SSH and SNMP protocols only.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>We select the MID Server from Domain A - in this case <strong>sandb01-358</strong>.</td>
</tr>
<tr>
<td>Active</td>
<td>Make sure this check box is selected to enable this behavior.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Leave the default criteria of Any. Criteria are not used for non-WMI functionalities.</td>
</tr>
</tbody>
</table>
Functionality 3: WMI Scanning on Domain B

All that remains is to create a functionality for the WMI scans on Domain B. Because of the Windows authentication mechanism, we must configure a Windows MID Server to scan Domain B that is a member of that domain.

Create the third functionality using the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field. All functionalities in this example use the same phase number, which launches a single Shazzam probe for all the functionalities in that phase. A single phase, when feasible, is the most efficient use of the Shazzam probe.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select Windows, DNS, and WINS from the list. This functionality defines the WMI protocol that will be scanned and resolves the domain. Because we selected to scan for WMI, we must select a Windows MID Server for this functionality.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>We select a Windows MID Server from Domain B - in this case disco-win2003.</td>
</tr>
<tr>
<td>Active</td>
<td>Make sure this check box is selected to enable this behavior.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Change the criteria to All.</td>
</tr>
</tbody>
</table>

Create Functionality Criteria

All Windows functionality requires criteria to identify the Windows domain and the MID Server. In our example, we will create two criteria for this functionality. To create Functionality Criteria, click New in the Related List and enter the following values for the MID Server doing the WMI scanning on Domain B:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Create the following criteria:</td>
</tr>
<tr>
<td></td>
<td>• Enter mid_server to name the MID Server that will execute the WMI scans on Domain B.</td>
</tr>
<tr>
<td></td>
<td>• Enter win_domain to name the Windows domain that Discovery will scan with the MID Server defined.</td>
</tr>
<tr>
<td>Operator</td>
<td>Select equals as the operator in this criteria.</td>
</tr>
<tr>
<td>Value</td>
<td>• For the mid_server value, enter the name of the MID Server that will scan Domain B for Windows devices.</td>
</tr>
<tr>
<td></td>
<td>• For the win_domain value, enter the name of Domain B that this MID Server will scan for Windows devices.</td>
</tr>
<tr>
<td>Active</td>
<td>Be sure to enable the criteria by selecting this check box (true).</td>
</tr>
</tbody>
</table>
The completed criteria appear in the Discovery Functionality form for this behavior.

**Discovery behavior example: access an ACL protected SNMP device**

This topic explains how to access an SNMP device protected by an ACL using a Discovery behavior.

**Role required:** admin

A specific type of Access Control List (ACL) on an SNMP device can prevent Discovery from identifying that device. This list defines host machines by IP addresses that are permitted to run agents on SNMP devices. In this example, we want to classify, identify, and update all the devices in an IP range, including the SNMP devices protected by an ACL. To do this, we must install a MID Server with access to the protected SNMP devices and then create a Behavior that allows us to scan for all protocols without missing any devices or doing any extra work.

1. **Install a MID Server on a host permitted by the ACL.**

   To scan the SNMP devices in a network protected by an ACL, a MID Server must be installed on a host machine specified by IP address in the ACL as having access to the SNMP devices. Because this MID Server is scanning SNMP devices only, it can be installed on any supported operating system. The other MID Server, configured to discover WMI and SSH, can be installed on any Windows host that has visibility to the specified IP ranges.

2. **Install additional MID Servers if you intend to configure either functionality to load balance.**

3. **Create the behavior record.**

   a) Create a new Discovery Behavior record, and name it something descriptive, such as Southwest-SNMP.

   b) Right-click in the header bar and select **Save** from the pop-up menu.

      This action saves the record and creates the Discovery Functionality related list in the form.

   c) To complete this behavior (as shown in **Figure 1**) we must create functionalities using the MID Servers we installed for that purpose.

      For the functionality that scans for the WMI protocol, we must define criteria that specify the Windows domain and MID Server being used. Because this Discovery is being performed on one Windows domain, we can configure a functionality for WMI and SSH (**All except SNMP**) using the same MID Server.

<table>
<thead>
<tr>
<th>Discovery Functionality</th>
<th>Name</th>
<th>Discovery Behavior = Southwest-SNMP</th>
</tr>
</thead>
<tbody>
<tr>
<td># Phase 1</td>
<td># MID Servers</td>
<td># Functionality</td>
</tr>
<tr>
<td>1</td>
<td>discover2003</td>
<td>All except SNMP</td>
</tr>
<tr>
<td>1</td>
<td>sassa2012_Softnet</td>
<td>SNMP only</td>
</tr>
</tbody>
</table>

**Completed Discovery Behavior**

4. **Define the functionalities.**
For this behavior, we create two functionalities, one for SNMP on the MID Server installed on the host specified in the ACL and another for WMI and SSH on a second MID Server.

a) Click **New** in the Discovery Functionality related list to add functionalities.

b) **Note:** We configure the MID Server installed on the privileged host machine to scan for the SNMP protocol only.

Create the first functionality using the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field. Both functionalities in this example use the same phase number, which launches a single Shazzam probe for all the functionalities in that phase. A single phase, when feasible, is the most efficient use of the Shazzam probe.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select SNMP only from the list. This functionality defines the protocol that will be scanned.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>We select a MID Server (sansol02_Solaris) installed on a Solaris host that is granted access to the SNMP devices by the ACL. You can select multiple MID Servers if you want Discovery to load balance this functionality automatically.</td>
</tr>
<tr>
<td>Active</td>
<td>Make sure this check box is selected to enable this behavior.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Leave the default criteria of Any. Criteria are not used for SNMP functionalities.</td>
</tr>
</tbody>
</table>

**Scan for SNMP Only**

c) **Note:** In the second functionality, Discovery will scan our domain for WMI and SSH protocols. Because WMI is one of the protocols, the MID Server for this functionality must be installed on a Windows machine and must have criteria.
Create the second functionality using the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Type a phase number of 1 in this field. All functionalities in this example use the same phase number, which launches a single Shazzam probe for all protocol scans. A single phase, when feasible, is the most efficient use of the Shazzam probe.</td>
</tr>
<tr>
<td>Functionality</td>
<td>Select All except SNMP from the list. This functionality defines the protocols for which scanning will occur. Because WMI is one of the protocols selected, we must use a Windows MID Server for this functionality.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>We select a Windows MID Server from our domain. In this case we select disco-win2003. You can select multiple MID Servers if you want Discovery to load balance this functionality automatically.</td>
</tr>
<tr>
<td>Active</td>
<td>Make sure this check box is selected to enable this behavior.</td>
</tr>
<tr>
<td>Match criteria</td>
<td>Change the criteria to All.</td>
</tr>
</tbody>
</table>

**Note:** Functionalities that scan for WMI require criteria to identify the domain and the MID Server. In our example, we will create two criteria for this functionality.

To create functionality criteria for a WMI scan, click **New** in the related list.

**e)** Enter the following values for the MID Server doing the WMI scanning on our domain:
<table>
<thead>
<tr>
<th>Field</th>
<th>Input Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Create the following criteria:</td>
<td>· Enter <code>mid_server</code> to name the MID Server that will execute the WMI scans.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· Enter <code>win_domain</code> to name the Windows domain that Discovery will scan with the MID Server defined.</td>
</tr>
<tr>
<td>Operator</td>
<td>Select equals as the operator in this criteria.</td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td></td>
<td>· For the <code>mid_server</code> value, enter the name of the MID Server that will scan our domain for Windows devices.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· For the <code>win_domain</code> value, enter the name of our Windows domain.</td>
</tr>
<tr>
<td>Active</td>
<td>Be sure to enable the criteria by selecting this check box (<code>true</code>).</td>
<td></td>
</tr>
</tbody>
</table>

The completed criteria appear as follows in the Discovery Functionality form for this behavior.

Functionality Criteria for a WMI Scan

5. Create the schedule.
   a) Create a Discovery Schedule record and name the behavior we just created.
   b) Navigate to Discovery > Discovery Schedules and click New in the record list.
   c) Select a Discover mode of Configuration items. This action displays the Behavior field.
   d) Click the magnifier icon and select the behavior to use. In our example, we select Southwest-SNMP.
e) Select the appropriate time to run Discovery for this location.

f) Click **Quick Ranges** and define the IP address ranges, networks, or lists to scan for this domain.

g) Save the record.

---

**Discovery Schedule Record**

**CI deletion strategies for pattern discovery**

When you perform discovery with a pattern, you can choose what to do with CIs that are in the CMDB but Discovery can no longer find.

**Note:** Deletion strategies are available only with patterns, not probes/sensors.

When you use a pattern to perform discovery, you identify a CI Type, which is the main CI you are trying to find, such as an IIS web server or Tomcat server. Discovery can also find related CIs and populate the CMDB accordingly. For example, after Discovery finds an IIS server with the IIS pattern, it can also find these related CIs:

- The Windows machine that hosts the web server
- The configuration files for the IIS server
The web site that the IIS server hosts

The dependency map on the main CI also shows the related CIs, as illustrated by the following graphic:

An example of an IIS server main CI and its related CIs

After you discover the main CI and its related CIs for the first time with a pattern, Discovery puts an entry into the Related CI Types (sa_ci_to_pattern) table for each relation between the main CI and a related CI type. These records are where you determine the deletion strategy for each related CI by the type to which it belongs. You cannot set a deletion strategy for the main CI, except for AWS and Azure cloud databases.

You can view a list of the relations as the following image illustrates:
Example CIs related to an IIS server

The deletion strategy on a related CI takes effect when both of these conditions occur:

- After you run Discovery with the pattern again.
- Discovery successfully finds the main CI but cannot find the related CI.

These deletion strategy options are available:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keep</td>
<td>Preserves the configuration file CI record and makes no other changes to the record. This option is the default setting.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the configuration file CI record from the CMDB, and the relationship to the main CI.</td>
</tr>
<tr>
<td>Mark as absent</td>
<td>Marks the Status (install_status) field of the configuration file CI record as Absent, meaning that Discovery cannot find the tracked configuration file. This status does not instruct the system to delete the actual CI or the relationship.</td>
</tr>
</tbody>
</table>

**Note:** There are two tables that do have their CIs deleted if marked as absent: cmdb_ci_network_adapter and cmdb_ci_ip_address.

| Delete relations | Deletes only the CI relationships between the related CI and the main CI. |

Set a deletion strategy

Set a deletion strategy when you want to take action on a related CI that Discovery can no longer find through pattern discovery. You can delete the main CI only when it is an AWS or Azure cloud database.

- Role required: cloud_admin, sm_admin, or admin
- CIs: a main CI and related CIs that the discovery process found through pattern discovery
You can set a deletion strategy for all discovered CIs, including configuration file CIs. However, the way you access the deletion strategy for tracked configuration files differs from the procedure in this topic. See Set the deletion strategy for tracked configuration files for instructions.

1. Open the list of related CI types by entering `sa_ci_to_pattern.list` in the application filter.
2. Filter the list to find the CI type that is related to a main CI through pattern discovery. If you want to set a deletion strategy for a cloud database, search the list for `cmdb_ci_cloud_database` CI type.

Notice that the `cmdb_ci_cloud_database` CI types have the `Is Main CI` value set to `true`. This indicates that you can set a deletion strategy for both AWS and Azure databases, even if they are the main CI that cloud discovery finds.

3. From the list view, double click the value in the field in the `Deletion Strategy` column.
4. Select a new value:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keep</td>
<td>Preserves the configuration file CI record and makes no other changes to the record. This option is the default setting.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the configuration file CI record from the CMDB, and the relationship to the main CI.</td>
</tr>
<tr>
<td>Mark as absent</td>
<td>Marks the <code>Status (install_status)</code> field of the configuration file CI record as <code>Absent</code>, meaning that Discovery cannot find the tracked configuration file. This status does not instruct the system to delete the actual CI or the relationship.</td>
</tr>
</tbody>
</table>

**Note:** There are two tables that do have their CIs deleted if marked as absent: `cmdb_ci_network_adapter` and `cmdb_ci_ip_address`.

5. Select a new value:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete relations</td>
<td>Deletes only the CI relationships between the related CI and the main CI.</td>
</tr>
</tbody>
</table>

### Configuration file tracking

The horizontal discovery process can find configuration files that belong to certain applications and add those configuration files to the CMDB. You can track the changes to these files by comparing them to previous versions.

**Warning:** Configuration files contain sensitive system information. To prevent unauthorized access, ensure that access control lists (ACL) are placed on the Tracked Configuration file table (`cmdb_ci_config_file_tracked`). Only allow authorized users to view this table or uncheck the `Save Content` setting.
Components for configuration file tracking

CI type

All applications and hosts in your organization must have a corresponding configuration item (CI) type, which is necessary for discovering and processing applications and hosts correctly. In a base system, many CI types have configuration file paths defined for them. You can add new or modify existing definitions for tracking configuration files. See Modify tracking changes in configuration files for instructions.

Patterns

Configuration file tracking is available for patterns that discover applications. On the pattern, you can create tracked file definitions that specify the CI type to which the application CI belongs and the path of the configuration file. Specify as many tracked file definitions as needed. You can also specify whether you want to save the contents of configuration files so you can view and compare the contents of different versions.

Note:

Configuration file tracking is not available for discoveries performed by traditional probes and sensors.

The classifier that triggers the pattern must specify the Horizontal Pattern probe, which in turn, must specify the pattern. If you upgrade your instance to the current version, not all classifiers are configured to use patterns for discovery by default.

CMDB

All configuration files are saved as a CI in the Tracked Configuration file (cmdb_ci_config_file_tracked) table. If you enable the content to be saved, these CI records provide the contents of the configuration files, including previous versions. From the configuration file CI record, you can compare different versions.

Dependency maps and application service maps

Both dependency maps and application service maps display tracked configuration files. The relationship between a configuration file and its host is a contains relationship. The application contains the configuration file.

For example, this IIS web server contains three tracked configuration files:
Sometimes you organize CI types as a main CI type and its related CI types. On an application service map, Service Mapping shows changes to configuration files of related CIs for the main CIs in inclusions. In inclusions, the system treats applications hosted on a server as independent objects. For example, the Tomcat WAR CI appears separate from its host, the Tomcat CIs. In this case, Service Mapping shows changes to configuration files of Tomcat WAR when you select Tomcat. In addition, Service Mapping displays changes to configuration files of the hardware server hosting inclusions. In this example, it is a Linux server:

**Deletion strategy**

You can specify what you want to do with tracked configuration file CI records when discovery can no longer find them. You can keep the configuration file CI record, automatically delete it, delete only the CI relationships to it, or mark it absent.

**Discovery patterns that support configuration file tracking by default**

These patterns provide tracked file definitions by default:
<table>
<thead>
<tr>
<th>Classifier</th>
<th>Pattern</th>
<th>CI Type</th>
<th>File path of tracked file</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Apache Server</strong></td>
<td>Apache On Unix Pattern</td>
<td><strong>Apache Web Server</strong></td>
<td>$config_file</td>
</tr>
<tr>
<td></td>
<td>Apache On Windows Pattern</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MySQL Server</strong></td>
<td>MySQL server On Windows and Linux Pattern</td>
<td><strong>MySQL Instance</strong></td>
<td>$config_file</td>
</tr>
<tr>
<td><strong>Microsoft IIS Server</strong></td>
<td>IIS</td>
<td><strong>Microsoft IIS Web Server</strong></td>
<td>EVAL(javascript: var rtrn = '';var winDir = CTX.getCommandManager().shellCommand(&quot;echo %WinDir%&quot;, false, null, null, CTX);rtrn = winDir.trim() + '\\System32\inetsrv\Config*.config&quot;);) IIS Virtual Directory</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Active Matrix Business Works</strong></td>
<td>Active Matrix Business Works</td>
<td><strong>ActiveMatrix Business Works</strong></td>
<td>$config_file</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Enterprise Message Service</strong></td>
<td>Enterprise Message Service</td>
<td><strong>Tibco Enterprise Message Service</strong></td>
<td>$config_file</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Oracle</strong></td>
<td>Oracle DB on Windows Pattern</td>
<td><strong>Oracle Instance</strong></td>
<td>$install_directory + &quot;\network\admin*.ora&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Oracle Instance</strong></td>
<td>$install_directory + &quot;\dbs*.ora&quot;</td>
</tr>
<tr>
<td></td>
<td>Oracle DB on Unix Pattern</td>
<td><strong>Oracle Instance</strong></td>
<td>$install_directory + &quot;/dbs/<em>.</em>.ora&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Oracle Instance</strong></td>
<td>$install_directory + &quot;<em>/network/admin/</em>.*.ora&quot;</td>
</tr>
<tr>
<td><strong>Tomcat</strong></td>
<td>Tomcat</td>
<td><strong>Tomcat</strong></td>
<td>$install_directory + &quot;*/conf/*server.xml&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Tomcat WAR</strong></td>
<td>$install_directory + &quot;*/WEB-INF/*web.xml&quot;</td>
</tr>
<tr>
<td><strong>WMB</strong></td>
<td>WMB On Unix Pattern</td>
<td><strong>IBM WebSphere Message Broker</strong></td>
<td>$install_directory + &quot;<em>/etc/config/</em>/*.prop&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>IBM WebSphere Message Broker</strong></td>
<td>$install_directory + &quot;<em>/etc/config/</em>/*.prop&quot;</td>
</tr>
<tr>
<td></td>
<td>WMB On Windows Pattern</td>
<td><strong>IBM WebSphere Message Broker</strong></td>
<td>$install_directory + &quot;<em>/etc/config/</em>/*.prop&quot;</td>
</tr>
<tr>
<td><strong>WMQ</strong></td>
<td>WMQ On Windows Pattern</td>
<td><strong>IBM WebSphere MQ</strong></td>
<td>$install_directory + &quot;<em>/etc/config/</em>/*.sh&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>IBM WebSphere MQ</strong></td>
<td>$install_directory + &quot;<em>/etc/config/</em>/*.sh&quot;</td>
</tr>
<tr>
<td></td>
<td>WMQ On Windows Pattern</td>
<td><strong>IBM WebSphere MQ</strong></td>
<td>$install_directory + &quot;<em>/bin/</em>.*.sh&quot;</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
What to do

1. **Enable configuration file tracking** by setting the glide.discovery.enable_file_tracking Discovery property to true.
   
   You can also configure other properties to control the size and number of tracked configuration files, the time window during which changes to configuration files are tracked for a given version, and the number of changes allowed on a configuration file during that time window. See [Discovery properties](#) for details.

2. **To prevent unauthorized access to the sensitive information from configuration files,** perform the following steps:
   
   a. Ensure that access control rules (ACL) are placed on the Tracked Configuration file table (cmdb_ci_config_file_tracked). Only allow authorized users to view this table or uncheck the **Save Content** setting.
   
   b. Ensuring that the tracked_file_reader role that controls access to configuration file data is assigned to correct users and user groups. By default, the itil role contains the tracked_file_reader role giving all users with the itil role access to the configuration file information.

3. **Verify that the Horizontal Discovery probe is active on the classifier for the software** that you want to discovery. If not, you can enable it, specify the pattern, and then disable the other probes. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

4. **Add or modify tracked file definitions** to change the CI type or file path.

5. **Set the tracked files deletion strategy** to specify what you want to do with tracked configuration file CI records when pattern discovery can no longer find them.

6. **Run horizontal discovery** on the hosts that are running the applications you want to discover with patterns, open the application CI record, and check the Tracked Configuration Files related list.

7. **Compare two versions** of tracked CI configuration files to see the actual changes made to them.

**Modify tracking changes in configuration files**

Configure the system to collect information about changes in configuration files belonging to a configuration item (CI). Service Mapping uses this information to notify users that CI configuration files changed and to view actual changes to configuration files directly in the application service maps.

Role required: cloud_admin, sm_admin or admin

To enable tracking configuration files for a CI for which the system does track files, configure classification for the CI type to which this CI belongs:

- For CI types representing applications, [create a Discovery process classification](#) for the relevant pattern. Add the **Horizontal Pattern** probe in the **Probe** column, and then specify your pattern in the **Pattern** column.
- For CI types representing SNMP devices, perform configuration as described in [Create a Discovery CI classification](#).

**Note:** There is no need to create CI classifications for hosts because these classifications are included in the base system.
The system tracks configuration files as part of horizontal discovery process of a CI to which these files belong. Configuration files contain CI settings and parameters. Service Mapping uses this information to notify users that CI configuration files changed and to view actual changes to configuration files directly in the application service maps. To learn about how the system tracks configuration files, stores and updates information about these files, see Configuration file tracking.

The system tracks configuration files for CI types with process classifiers that trigger patterns. The patterns contain tracked file definitions, which specify the CI type that the application belongs to and the path of the configuration file. Several patterns, such as the IIS pattern, provide tracked file definitions by default. You can use these default definitions or create your own for any pattern a process classifier triggers.

1. Navigate to Pattern Designer &gt; Discovery Patterns.
2. Click the pattern for the relevant CI.

**Attention:** Enabling configuration file tracking does not modify the contents of a pattern.

3. Click the Tracked Files tab.

One or more tracked file definitions appear by default for Discovery patterns that support configuration file tracking.

If you do not see the Tracked Files tab, verify that the pattern is specified process classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

4. Click **Edit** to modify an existing definition or click **New** to create a new one.
5. Fill out or modify the form fields (see table):

---

**Related Links**

Edit deletion strategy for the configuration files of this pattern

---

Tracked files on the IIS pattern
**Tracked File Definition**

Define the file path for the selected CI type. File path consists of CI attributes in variable or wildcard format. For example, $install_directory+"/conf/*.xml"

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Select either the primary or related CI type to which the configuration file belongs.</td>
</tr>
<tr>
<td>File Path</td>
<td>Define the path of the configuration file to track. You can use pattern variables and wildcards. For example, for the following path: &lt;the folder with the &quot;install_directory&quot; parameter&gt;/conf/&lt;any xml file&gt;, enter $install_directory+&quot;/conf/*.xml&quot;</td>
</tr>
<tr>
<td>Save Content</td>
<td>Select this check box to make this file available for viewing and comparing directly the ServiceNow instance.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this check box to enable tracking of this configuration file.</td>
</tr>
</tbody>
</table>

6. Click **Done**.
7. If necessary, set the tracked files deletion strategy as described in [Set the deletion strategy for tracked configuration files](#).

Run discovery on the hosts that are running the applications you want to discover with patterns, open the application CI record, and check the Tracked Configuration Files related list.
If you are using Service Mapping, verify that it started collecting information about changes to configuration files:

1. Navigate to the application service map containing the CIs.
2. On the Changes tab, check that the list includes a record of the change you made. For example, that the configuration file was added, modified or deleted.
Compare versions of CI configuration files

You can compare two versions of tracked CI configuration files to see the actual changes made to them.

If Discovery and Service Mapping do not track a configuration file you want to track, perform advanced configuration as described in Modify tracking changes in configuration files.

Role required: pd_admin or pd_user

Discovery and Service Mapping check configuration files for changes during horizontal discovery using patterns. If a configuration file is modified, the system saves the version of the file. You can use the timestamp to differentiate between versions.

Changes to configuration files are associated with CIs to which these files belong. Maps show configuration file changes as changes to related CIs.

1. To compare the latest and the previous version of the CI configuration file from Service Mapping:
   a) Navigate to Service Mapping > Services > Business Services.
   b) Click View Map next to the required business service.
   c) On the Changes tab, locate the change record for the modified CI configuration file.

   **Important:** The name of the change must be File Modified.

   d) Right-click the change record for the modified CI configuration file, and select View Change in Tracked File Content.

   The Tracked Configuration File Comparison tab opens, displaying the most recent and the previous versions of the configuration file.

   e) Continue with 3.

2. To compare any two versions of the CI configuration file from the list of versions from Discovery:
   - Navigate to Discovery > Tracked Configuration Files and sort or filter the list by criteria such as Related CI or Host.
   - Open a CI record from the Configuration module for the CI type to which the configuration file belongs. For example, to find a Tomcat server, navigate to Configuration > Applications Servers > Tomcat, then select the specific Tomcat server. The list of configuration files appear in the Tracked Configuration files related list.

   a) Select the configuration file from the list.

   The tracked configuration file record shows:
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Name</td>
<td>The name and file path of the file.</td>
</tr>
<tr>
<td>File Size</td>
<td>The file size.</td>
</tr>
<tr>
<td>Last Modified</td>
<td>The date the file was last modified.</td>
</tr>
<tr>
<td>File Content</td>
<td>The contents of the file, if you selected the option to save the contents on the file tracking definition.</td>
</tr>
</tbody>
</table>

b) In the **Content Versions** related list, click one of the versions in the **Update time** related list. The same information about the configuration file appears along with the update time.

c) Click **Compare**.

The Tracked Configuration File Comparison tab opens, displaying the configuration file versions you selected.

3. Review actual changes.

Highlight colors indicate the type of change:
- Purple — Updated line
- Pink — New line
- Gray — Deleted line

4. Navigate between the changes using the arrows in the upper right corner.

*Set the deletion strategy for tracked configuration files*

You can specify what you want to do with tracked configuration file CI records when pattern discovery can no longer find them.

- Role required: discovery_admin, sm_admin, or admin
- A pattern that specifies tracked configuration files.

You can set a deletion strategy for all discovered CIs, not just configuration file CIs. However the way you access the deletion strategy for standard CIs differs from configuration file CIs. See **Set a deletion strategy** for instructions.

1. Navigate to **Pattern Designer > Discovery Patterns** and open the desired pattern.
2. Click the **Tracked Files** tab.
3. Under **Related Links**, click **Edit deletion strategy for the configuration files of this pattern**.
   The Related CI Types list appears showing you filtered records in the Related CI Type (sa_ci_to_pattern) table. The filter shows you only records with the CI type of Tracked Configuration files (cmdb_ci_config_file_tracked), which is the table where the system saves configuration file CIs.
4. From the list view, double-click the value in the field in the **Deletion Strategy** column for the tracked configuration file.
5. Select a new value:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keep</td>
<td>Preserves the configuration file CI record and makes no other changes to the record. This option is the default setting.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the configuration file CI record from the CMDB, and the relationship to the main CI.</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Mark as absent</td>
<td>Marks the Status (install_status) field of the configuration file CI record as Absent, meaning that Discovery cannot find the tracked configuration file. This status does not instruct the system to delete the actual CI or the relationship.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> There are two tables that do have their CIs deleted if marked as absent: cmdb_ci_network_adapter and cmdb_ci_ip_address.</td>
</tr>
<tr>
<td>Delete relations</td>
<td>Deletes only the CI relationships between the related CI and the main CI.</td>
</tr>
</tbody>
</table>

6.

**Change the source name of Discovery results**

You have the option of changing the source name of discovery results. This might be desirable if Discovery is running on your network together with another discovery product, and you want to use customized identifiers.

Role required: admin

The Source (sys_object_source) table stores information identifying the source of a discovery, the ID of that source, and the date/time of the last scan.

To view this information, configure a CI form and add the **Sources** related list. This table is populated automatically when the Discovery plugin is enabled.

---

**Sources List**

To migrate your Discovery Source to ServiceNow:

1. Update the **glide.discovery.source_name** system property to have a value of ServiceNow.
2. Edit the dictionary entry for the Configuration Item (cmdb_ci) table.
3. Update the choice list for the discovery_source column to only have the value ServiceNow for a discovery product.
4. Open a list for the Configuration Item (cmdb_ci) table and filter on a discovery source field value you want to change.
5. Use the Update All option to change the value to ServiceNow.
6. Open a list for the Source (sys_object_source) table and filter on a name field value you want to change.
7. Use the Update All option to change the value to ServiceNow.

**Discovery API plugin**

The Discovery API plugin provides APIs for scoped applications and is loaded when the Discovery plugin is activated.

Details about these Discovery API methods are available on the ServiceNow Developer Portal. They are listed here by class.

**Note:** Java API methods are not customizable.

**DiscoveryAPI - Scoped**

The methods in this class launch a quick Discovery of a single IPv4 address and return summaries of previously launched Discovery statuses for a single CI or for all scanned CIs. A MID Server is selected automatically, based on the IP address provided or the application specified.

- **discoverIpAddress()**: Discovers a single IPv4 address.
- **reportCiIpAddressStatus()**: Returns a summary of a configuration item’s Discovery status given the specific status sys_id and IPv4 address.
- **reportCiStatus()**: Returns a summary of a CI Discovery status given a specific Discovery Status sys_id.

**ReportCiStatusOutputJS**

The methods are getters that return specific object properties for the DiscoveryAPI reportCiIpAddressStatus method and then convert the information into a JSON string.

- **getCiOperationStatus()**: Used to return the state of the scanned CI.
- **getCmdbCi()**: Used to return the value in the cmdb_ci field from the discovery_device_history table for the CI being scanned.
- **getDiscoveryState()**: Used to return the value from the State field in the Discovery Status (discovery_status) table.
- **getIpAddress()**: Used to return the value from the source field in the discovery_device_history table for the CI being scanned.
- **getIssues()**: Used to return the value from the issues field in the discovery_device_history table for the CI being scanned.
- **getIssuesLink()**: Used to return the value from the issues_link field in the discovery_device_history table for the CI being scanned.
- **toJson()**: Used to serialized the ReportCiStatusOutputJS object.

**Data collected by Discovery**

Discovery collects unique data for each type of device and stores it in dedicated tables, fields, and relationships.
Computers and servers
Find servers and computers based on the machine's operating system:
- Windows
- Solaris
- Linux
- OS/X
- HPUX
- AIX
- VMware virtual machines
- Hyper-V
- JBOSS on Windows and Linux

Network devices
Detect several types of network-related CIs:
- Load balancers
- Routers and switches
- IP networks and specific IP addresses
- Cisco UCS devices
- Printers and power supplies
- IP services and daemons
- IBM Websphere datapower devices

You can also find devices based on TCP connections and Layer-2, SNMP-level discovery.

Software
Find a variety of software, including:
- Web and email servers, including Microsoft IIS, Apache, Tomcat, Exchange mailbox
- Puppet automation software
- WebLogic and WebSphere servers
- Clustered applications
- Tibco, Microsoft, IBM, HP, and SAP applications
- General software

Databases
Discover databases, including:
- Oracle
- MySQL
- MSSQL
- MongoDB
- SAP HANA
- PostgreSQL discovery
- Sybase discovery
**Storage**

Discover these types of storage devices:
- Direct Attached Storage (DAS)
- Storage Area Networks (SAN)
- Network Attached Storage (NAS)
- Host bus adapters (HBA)

**AWS and Azure resources**

Discover resources in Amazon and Azure clouds:
- Datacenters
- Virtual machines
- Availability zones
- Cloud networks and subnets
- Network interfaces and load balancers

**Operating system level virtualization**

Collect OSLV information:
- Containers, engines, images, and tags
- Docker components

**Computer and server discovery**

Discovery identifies the following computers, clusters, and virtual machines.

**AIX server discovery**

Discovery identifies and classifies information about AIX servers.

**Credentials**

Configure **SSH credentials**.

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.
Classifier, probes, and pattern

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX</td>
<td>Horizontal discovery probe: launches patterns</td>
<td>AIX server</td>
</tr>
</tbody>
</table>
| (UNIX Classification) | AIX - Installed Software^  
|                 | AIX - ADM^  
|                 | AIX - Identity*  
|                 | AIX - CPU*  
|                 | AIX - Hardware Model*  
|                 | AIX - Memory*  
|                 | AIX - OS Patches*  
|                 | AIX - OS Release*  
|                 | AIX - OS Revision*  
|                 | UNIX - Find FQDN  
|                 | UNIX - OS Filesystems  
|                 | UNIX - OS Uptime  |

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^These probes remain active by default, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_computer</td>
<td>os</td>
<td>uname</td>
</tr>
<tr>
<td>OS version</td>
<td>cmdb_ci_computer</td>
<td>os_version</td>
<td>oslevel</td>
</tr>
<tr>
<td>OS service pack</td>
<td>cmdb_ci_computer</td>
<td>os_service_pack</td>
<td>oslevel</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_aix_server</td>
<td>short_description</td>
<td>uname</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_aix_server</td>
<td>name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_aix_server</td>
<td>host_name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_aix_server</td>
<td>dns_domain</td>
<td>DNS</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci_aix_server</td>
<td>start_date</td>
<td>uptime</td>
</tr>
<tr>
<td>CPU type</td>
<td>cmdb_ci_computer</td>
<td>cpu_type</td>
<td>lsdev, lsattr</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_computer</td>
<td>cpu_speed</td>
<td>lsdev, lsattr</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_computer</td>
<td>cpu_count</td>
<td>lsdev, lsattr</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_aix_server</td>
<td>manufacturer</td>
<td>lsattr</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_aix_server</td>
<td>model_id</td>
<td>lsattr</td>
</tr>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_computer</td>
<td>ram</td>
<td>lsdev, lsattr</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_file_system</td>
<td>name</td>
<td>df</td>
</tr>
</tbody>
</table>
### Label | Table Name | Field Name | Source  
--- | --- | --- | ---  
Capacity (MB) | cmdb_ci_file_system | capacity | df  
Available Space (MB) | cmdb_ci_file_system | available_space | df  
Mount point | cmdb_ci_file_system | mount_point | df  
Name | cmdb_ci_patches | name | instfix  
Name | cmdb_running_process | name | ps  
Command | cmdb_running_process | command | ps  
Connects to | cmdb_running_process | connects_to | lsof  
Listening on | cmdb_running_process | listening_on | lsof  
Type | cmdb_running_process | type | ps  
PID | cmdb_running_process | pid | ps  
Parameters | cmdb_running_process | parameters | ps  
Name | cmdb_ci_network_adapter | name | ifconfig, netstat  
IP address | cmdb_ci_network_adapter | ip_address | ifconfig, netstat  
MAC address | cmdb_ci_network_adapter | mac_address | ifconfig, netstat  
Netmask | cmdb_ci_network_adapter | netmask | ifconfig, netstat

**HBase instance discovery**

Discovery creates or updates a CMDB record when it detects a running instance of HBase on a UNIX server.

In the following table, the *HBase Instance@hostname* source may contain the information on whether the instance is a master or a slave. The name contains *Hmaster* if the instance is a master instance. The name contains *HRegionServer* if the name is a slave instance.

#### Data collected

| Label | Table Name | Field Name | Source  
--- | --- | --- | ---  
Name | cmdb_ci_db_hbase_instance | name | HBase Instance@hostname  
Root Directory | cmdb_ci_db_hbase_instance | root_dir | hbase-site.xml  
TCP port | cmdb_ci_db_hbase_instance | tcp_port | running process  
Site XML | cmdb_ci_db_hbase_instance | site_xml | hbase-site.xml  
Version | cmdb_ci_db_hbase_instance | version | HBase shell  
HBase Home | cmdb_ci_db_hbase_instance | hbase_home | running process  
ZooKeeper Quorum | cmdb_ci_db_hbase_instance | zookeeper | hbase-site.xml

**HPUX discovery**

Discovery identifies and classifies information about HPUX computers.

**Credentials**

Configure [SSH credentials](#).

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

### Classifier, probes, and pattern

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
</table>
| HP-UX (Unix classifier) | • Horizontal discovery probe: launches patterns  
                           • HP-UX - Installed Software^  
                           • HP-UX - ADM^  
                           • HP-UX - OS Patches*  
                           • HP-UX - CPU Speed*  
                           • HP-UX - Total Memory*  
                           • HP-UX - Hardware Model*  
                           • HP-UX - CPU Info*  
                           • HP-UX - Identity*  
                           • UNIX - Find FQDN*  
                           • UNIX - OS Filesystems*  
                           • UNIX - OS Uptime* | HP-UX Server |

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^These probes remain active by default, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

### Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_hpux_server</td>
<td>os</td>
<td>uname</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_hpux_server</td>
<td>short_description</td>
<td>uname</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_hpux_server</td>
<td>name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_hpux_server</td>
<td>host_name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_hpux_server</td>
<td>dns_domain</td>
<td>DNS</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci_hpux_server</td>
<td>start_date</td>
<td>uptime</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_computer</td>
<td>manufacturer</td>
<td>dmidecode</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_hpux_server</td>
<td>serial_number</td>
<td>uname</td>
</tr>
<tr>
<td>CPU type</td>
<td>cmdb_ci_hpux_server</td>
<td>cpu_type</td>
<td>cpuinfo</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_hpux_server</td>
<td>cpu_speed</td>
<td>adb</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_hpux_server</td>
<td>cpu_count</td>
<td>cpuinfo</td>
</tr>
</tbody>
</table>
Hyper-V discovery

Microsoft Hyper-V is a virtualization application that is included with the Windows Server 2008 operating system.

A physical machine running Hyper-V is divided into partitions (virtual machines), including a parent partition running Windows Server 2008 and child partitions running supported guests. The parent partition manages the virtual machines with the Hyper-V Manager application. On Windows Server 2008 this is done through the Microsoft Management Console (MMC) service. On Windows 7, use the Remote Server Admin tools.

Hyper-V supports the following functionality:

- **Failover clustering**: Failover is managed with Failover Cluster Manager.
- **Live migration**: Virtual machines can be moved between failover cluster nodes without bringing down the virtual machine.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Supported versions**

Discovery is supported for these Hyper-V Server versions:

- 2008
- 2012
- 2012 R2
- 2016
Important: Discovery of Hyper-V running on Windows 2016 is supported for patterns only. If you upgraded and are still using probes and sensors for Discovery, see KB0694773 for instructions on configuring Hyper-V probes.

Credentials

Configure Windows credentials with Domain administrator rights. You should also Enable PowerShell for the MID Server used to discover Hyper-V servers and instances.

Classifier, probes, and pattern

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hyper-V Server</td>
<td>• Horizontal discovery probe: launches patterns</td>
<td>Hyper-V Server</td>
</tr>
<tr>
<td></td>
<td>• Windows - ADM*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Windows - Installed Software*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Hyper-V - Cluster*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Hyper-V - Resource Pools*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Hyper-V - Virtual Machines*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Hyper-V - Virtual Networks*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Several other Windows probes also run, such as</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Identity, Windows - software, and so on.*</td>
<td></td>
</tr>
</tbody>
</table>

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^These probes remain active by default, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Data collected

Virtual instances

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Data Description</th>
<th>Stored in Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object ID^</td>
<td>object_id</td>
<td>ID</td>
<td>cmdb_ci_hyper_v_instance</td>
</tr>
<tr>
<td>Name^</td>
<td>name</td>
<td>name</td>
<td>cmdb_ci_hyper_v_instance</td>
</tr>
<tr>
<td>Label</td>
<td>Field Name</td>
<td>Data Description</td>
<td>Stored in Table</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------</td>
<td>------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
<td>• On</td>
<td>cmdb_ci_hyper_v_instance*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Off</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Suspended</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Changing</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Stuck</td>
<td></td>
</tr>
<tr>
<td>CPUs</td>
<td>cpus</td>
<td>Count</td>
<td>cmdb_ci_hyper_v_instance*</td>
</tr>
<tr>
<td>Memory</td>
<td>memory</td>
<td>Quantity in MB</td>
<td>cmdb_ci_hyper_v_instance*</td>
</tr>
<tr>
<td>Network adapters</td>
<td>nics</td>
<td>Count</td>
<td>cmdb_ci_hyper_v_instance*</td>
</tr>
<tr>
<td>Disks</td>
<td>disks</td>
<td>Count+</td>
<td>cmdb_ci_hyper_v_instance*</td>
</tr>
<tr>
<td>Disks size</td>
<td>disks_size</td>
<td>Capacity in GB+</td>
<td>cmdb_ci_hyper_v_instance*</td>
</tr>
<tr>
<td>Virtual Base Board Serial Number</td>
<td>baseboard_serial</td>
<td>Virtual serial number</td>
<td>cmdb_ci_hyper_v_instance</td>
</tr>
<tr>
<td>BIOS Serial Number</td>
<td>bios_serial</td>
<td>Virtual serial number</td>
<td>cmdb_ci_hyper_v_instance</td>
</tr>
<tr>
<td>Chassis Serial Number</td>
<td>chassis_serial</td>
<td>Virtual serial number</td>
<td>cmdb_ci_hyper_v_instance</td>
</tr>
<tr>
<td>BIOS GUID</td>
<td>bios_guid</td>
<td>Globally unique identifier (GUID)</td>
<td>cmdb_ci_hyper_v_instance</td>
</tr>
</tbody>
</table>

*Attributes that are put in the `cmdb_ci_hyper_v_instance` table when using the **Hyper-V Server** pattern. For upgrades, probes and sensors continue to put these attributes in the `cmdb_ci_vm_instance` table.

^Newly discovered attribute when using the **Hyper-V Server** pattern.

+Discovery can only return this information if the virtual machine is running.

### Virtual servers

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Stored in table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>Chassis type</td>
<td>chassis_type</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>Short description</td>
<td>short_description</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>OS Address Width (bits)</td>
<td>os_address_width</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>Operating System</td>
<td>os</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>Host name</td>
<td>hostname</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>OS Version</td>
<td>os_version</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>OS Service Pack</td>
<td>os_service_pack</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
<tr>
<td>Is virtual</td>
<td>virtual</td>
<td>cmdb_ci_hyper_v_server</td>
</tr>
</tbody>
</table>
Virtual networks

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Data Description</th>
<th>Stored in Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>Name of the virtual network</td>
<td>cmdb_ci_hyper_v_network*</td>
</tr>
<tr>
<td>ID</td>
<td>object_id</td>
<td>Globally unique identifier (GUID)</td>
<td>cmdb_ci_hyper_v_network*</td>
</tr>
</tbody>
</table>

*Attributes that are put in the cmdb_ci_hyper_v_network table when using the Hyper-V Server pattern. For upgrades, probes and sensors continue to put these attributes in the cmdb_ci table (name) and the cmdb_ci_vm_object table (object_id).

Resource pools

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Data Description</th>
<th>Stored in Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name*</td>
<td>name</td>
<td>Name of the resource pool</td>
<td>cmdb_ci_hyper_v_resource_pool</td>
</tr>
<tr>
<td>ID</td>
<td>object_id</td>
<td>Globally unique identifier (GUID)</td>
<td>cmdb_ci_hyper_v_rpool_comp</td>
</tr>
<tr>
<td>Capacity</td>
<td>capacity</td>
<td>Maximum amount of the appropriate allocation unit</td>
<td>cmdb_ci_hyper_v_rpool_comp</td>
</tr>
<tr>
<td>Allocation units</td>
<td>allocation_units</td>
<td>Units of measurement used (for example, MB or GB)</td>
<td>cmdb_ci_hyper_v_rpool_comp</td>
</tr>
<tr>
<td>Resource type</td>
<td>resource_type</td>
<td>Type of resource discovered (for example, memory, or hard drive space)</td>
<td>cmdb_ci_hyper_v_rpool_comp</td>
</tr>
</tbody>
</table>

Clusters

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Data Description</th>
<th>Stored in Table</th>
<th>Reference Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows Cluster</td>
<td>name</td>
<td>Reference field displaying the Windows Cluster on which the Hyper-V cluster resides.</td>
<td>cmdb_ci_hyper_v_cluster</td>
<td>cmdb_ci_hyper_v_machwin_cluster</td>
</tr>
</tbody>
</table>

Relationships

Discovery creates records in the CMDB for the host machine, the parent partition running Windows Server 2008, and all the virtual machines running on the Hyper-V server. Discovery finds the stored templates and maps all the relationships between components.

The instance modifies these tables for use with multiple virtualization products:
<table>
<thead>
<tr>
<th>Table</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtualization Server (cmdb_ci_virtualization_server)</td>
<td>Contains data on all discovered virtualization servers.</td>
</tr>
<tr>
<td>Virtual Machine Instance (cmdb_ci_vm_instance)</td>
<td>Contains data on all discovered virtual machine instances.</td>
</tr>
<tr>
<td>Virtual Machine Object (cmdb_ci_vm_object)</td>
<td>Contains data about various objects associated with a Hyper-V server, such as partitions, networks, resource pools, and clusters.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parent table</th>
<th>Relationship</th>
<th>Child table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hyper-V Virtual Machine Instance (cmdb_ci_hyper_v_instance)</td>
<td>Registered on::Has registered</td>
<td>Hyper-V Server (cmdb_ci_hyper_v_server)</td>
</tr>
<tr>
<td>Hyper-V Virtual Machine Instance (cmdb_ci_hyper_v_instance)</td>
<td>Connected by::Connects</td>
<td>Hyper-V Network (cmdb_ci_hyper_v_network)</td>
</tr>
<tr>
<td>Hyper-V Network (cmdb_ci_hyper_v_network)</td>
<td>Provided::Provides</td>
<td>Hyper-V Server (cmdb_ci_hyper_v_server)</td>
</tr>
<tr>
<td>Hyper-V Resource Pool (cmdb_ci_hyper_v_resource_pool)</td>
<td>Defines resources for::Gets resources from</td>
<td>Hyper-V Server (cmdb_ci_hyper_v_server)</td>
</tr>
<tr>
<td>Hyper-V Resource Pool Components (cmdb_ci_hyper_v_rpool_comp)</td>
<td>Contained by::Contains</td>
<td>Hyper-V Resource Pool (cmdb_ci_hyper_v_resource_pool)</td>
</tr>
<tr>
<td>Hyper-V Cluster (cmdb_ci_hyper_v_cluster)</td>
<td>Members::Member of</td>
<td>Hyper-V Server (cmdb_ci_hyper_v_server)</td>
</tr>
<tr>
<td>Hyper-V Server (cmdb_ci_hyper_v_server)</td>
<td>Runs on::Runs</td>
<td>Windows Servers (cmdb_ci_win_server)</td>
</tr>
<tr>
<td>Windows Servers (cmdb_ci_win_server)</td>
<td>Instantiates::Instantiated by</td>
<td>Hyper-V Virtual Machine Instance (cmdb_ci_hyper_v_instance)</td>
</tr>
<tr>
<td>Windows Servers (cmdb_ci_win_server)</td>
<td>Virtualized by::Virtualizes</td>
<td>Hyper-V Server (cmdb_ci_hyper_v_server)</td>
</tr>
</tbody>
</table>

**Clone Hyper-V virtual machines**

When importing (cloning) Hyper-V virtual machines, make sure each virtual machine has a unique ID. Discovery identifies virtual machines with duplicate IDs as the same machine.

1. Export the virtual machine from the Hyper-V server.
2. Import the virtual machine into a different server, making one of the these choices:
   - **Move or restore the virtual machine**: This selection clones the image, using the same ID, and moves it to another server. The original image remains on the Hyper-V server.
   - **Copy the virtual machine**: This selection copies the virtual machine, using a different ID, and moves it from the Hyper-V server. This is the best selection and does not require any further action before you run Discovery.
3. If you import the Hyper-V clone using the **Move or restore** selection, be sure to delete the original image from the Hyper-V server.
When Discovery encounters two virtual machines with the same equivalent serial numbers, it creates only one configuration item (CI).

**JBoss server discovery**

Discovery can detect JBoss application servers running on Linux and Windows systems.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Requirements

For JBoss application servers running on Linux systems.

- Set probe permissions to use these Bourne shell commands: `find`, `cat`, and `dirname`.
- Enable SSH on the JBoss application server. The SSH credential must also have read permissions on the `web.xml` and `jboss-service.xml` files.

For JBoss application servers running on Windows systems, enable *PowerShell* on the MID Server.

### Credentials

- [Applicative credentials](#)
- [Windows credentials](#)
- [SSH credentials](#)

For a list of privileged commands that you need for Discovery and Service Mapping, see *Service Mapping commands requiring a privileged user* for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

### Classifiers, patterns, and probes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>JBoss Server</td>
<td>• Horizontal Pattern: launches patterns</td>
<td>JBoss pattern</td>
</tr>
<tr>
<td></td>
<td>• Windows - JBoss Find web.xml List*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Windows - JBoss Get jboss-service.xml*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• JBoss - Find web.xml List*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• JBoss - Get jboss-service.xml*</td>
<td></td>
</tr>
</tbody>
</table>

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See *Add the Horizontal Pattern probe to a classifier* for instructions.
Data collected

Discovery populates the following fields in the JBoss (cmdb_ci_app_server_jboss) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Category</td>
<td>category</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
</tr>
</tbody>
</table>

Discovery populates the following fields on the Web Service (cmdb_ci_web_service) and Web Application (cmdb_ci_web_application) servers tables:

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ci_web_service</td>
<td>name</td>
<td>jboss-service.xml</td>
</tr>
<tr>
<td>App server</td>
<td>cmdb_ci_web_service</td>
<td>app_server</td>
<td>Internal reference</td>
</tr>
<tr>
<td>Description</td>
<td>cmdb_ci_web_application</td>
<td>short_description</td>
<td>web.xml</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_web_application</td>
<td>name</td>
<td>web.xml</td>
</tr>
<tr>
<td>Document base</td>
<td>cmdb_ci_web_application</td>
<td>document_base</td>
<td>web.xml</td>
</tr>
<tr>
<td>App server</td>
<td>cmdb_ci_web_application</td>
<td>app_server</td>
<td>web.xml</td>
</tr>
<tr>
<td>Servlet Name*</td>
<td>cmdb_ci_web_application</td>
<td>servlet_name</td>
<td>web.xml</td>
</tr>
<tr>
<td>Servlet Class*</td>
<td>cmdb_ci_web_application</td>
<td>servlet_class</td>
<td>web.xml</td>
</tr>
</tbody>
</table>

*Windows servers only

Relationships

<table>
<thead>
<tr>
<th>Parent class</th>
<th>Relationship</th>
<th>Child class</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_web_service</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_windows_server</td>
</tr>
<tr>
<td>cmdb_ci_app_server_jboss</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_web_application</td>
</tr>
<tr>
<td>cmdb_ci_app_server_jboss</td>
<td>Runs::Runs on</td>
<td>cmdb_ci_web_application</td>
</tr>
<tr>
<td>cmdb_ci_app_server_jboss</td>
<td>Runs::Runs on</td>
<td>cmdb_ci_web_service</td>
</tr>
</tbody>
</table>

Legacy: JBoss probes and sensors

Discovery identifies a Linux JBoss application server using probes and sensors.

Linux

This process describes the use of probes, which are active on your instance if you upgrade. For new instances, Discovery uses the JBoss pattern.
1. The UNIX - Active Processes probe detects a running process that matches an `org.jboss.main` entry point parameter.

2. If there is a match, a record is created in the JBoss Application Server (`cmdb_ci_app_server_jboss`) table. The following probes are also triggered:
   - JBoss - Find web.xml list: The sensor of this probe populates information in the Web Application (`cmdb_ci_web_application`) table if applicable.
   - JBoss - Get jboss-service.xml: The sensor of this probe populates information in the Web Service (`cmdb_ci_web_service`) table.

3. The JBoss - Find web.xml list probe searches for the `web.xml` files of JBoss application server. The probe uses the `classpath` parameter in the running process, and then searches in the related `server\default\deploy` directory for the JBoss installation.

4. If associated web applications reside in the `server\default\deploy` directory, the JBoss - Get web.xml probe triggers for each application. This probe reads the `web.xml` file for each web application and the sensor populates additional information to the Web Service (`cmdb_ci_web_service`) table.

5. The JBoss - Get jboss-service.xml probe uses the `classpath` parameter in the running process to search for the `jboss-service.xml` file in the related `server\default\conf\` directory for the JBoss installation.

6. If the probe successfully finds the `jboss-service.xml` file in the `server\default\conf\` directory, the sensor reads the contents of the XML file. It then creates additional records in the Web Service (`cmdb_ci_web_service`) table as necessary.

Windows

This process describes the use of probes, which are active on your instance if you upgrade. For new instances, Discovery uses the JBoss pattern.

1. The Windows - Active Processes probe detects a running process that matches an `org.jboss.main` entry point parameter.

2. If there is a match, a record is created in the JBoss Application Server (`cmdb_ci_app_server_jboss`) table. The following probes are also triggered:
   - JBoss - Find web.xml list: The sensor of this probe populates information in the Web Application (`cmdb_ci_web_application`) table if applicable.
   - JBoss - Get jboss-service.xml: The sensor of this probe populates information in the Web Service (`cmdb_ci_web_service`) table.

3. The JBoss - Find web.xml list probe searches for the `web.xml` files of JBoss application server. The probe uses the `classpath` parameter in the running process, and then searches in the related `server\default\deploy` directory for the JBoss installation.

4. If associated web applications reside in the `server\default\deploy` directory, the JBoss - Get web.xml probe triggers for each application. This probe reads the `web.xml` file for each web application and the sensor populates additional information to the Web Service (`cmdb_ci_web_service`) table.

5. The JBoss - Get jboss-service.xml probe uses the `classpath` parameter in the running process to search for the `jboss-service.xml` file in the related `server\default\conf\` directory for the JBoss installation.
6. If the probe successfully finds the `jboss-service.xml` file in the `server\default\conf\` directory, the sensor reads the contents of the XML file. It then creates additional records in the Web Service (cmdb_ci_web_service) table as necessary.

**Linux discovery**

Discovery identifies and classifies information about Linux computers.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Supported Linux Operating Systems**

These types of Linux OS's are supported:

- Red Hat
- Fedora
- Debian
- SUSE
- CentOS
- Ubuntu

**Required credentials**

Configure [SSH credentials](#).

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Classifier, probes, and pattern**

Discovery uses patterns to find several types of devices, operating systems, and applications. See [Pattern updates for the New York release](#) for more information.
<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
</table>
| Linux (Unix classification)) |  • Horizontal discovery probe: launches patterns  
• Linux - Installed Software^  
• Unix - ADM^  
• Linux - Identity*  
• Linux - CPU*  
• Linux - Distribution*  
• Linux - Find FQDN*  
• Linux - Memory*  
• Linux - Memory Modules*  
• Linux - Network ARP Tables*  
• Linux - Storage*  
• Linux - Amazon EC2*  
• Unix - OS Uptime*  
• UNIX - OS Filesystems*  
• UNIX - Find FQDN*  
• Unix - ADM Enhanced* | Linux Server |

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^These probes remain active by default, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

### Data collected

The **Linux** classifier triggers probes that perform the discovery. Several probes are launched during discovery. See the classifier for a list of the trigger probes.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_linux_server</td>
<td>os</td>
<td>uname</td>
</tr>
<tr>
<td>OS Version</td>
<td>cmdb_ci_computer</td>
<td>os_version</td>
<td>uname -a or cat /etc/*release</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_linux_server</td>
<td>short_description</td>
<td>uname</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_linux_server</td>
<td>name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_linux_server</td>
<td>host_name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_linux_server</td>
<td>dns_domain</td>
<td>DNS</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci_linux_server</td>
<td>start_date</td>
<td>uptime</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_computer</td>
<td>manufacturer</td>
<td>dmidecode</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_computer</td>
<td>serial_number</td>
<td>dmidecode</td>
</tr>
<tr>
<td>CPU type</td>
<td>cmdb_ci_linux_server</td>
<td>cpu_type</td>
<td>/proc/cpuinfo</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_linux_server</td>
<td>cpu_speed</td>
<td>/proc/cpuinfo</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_linux_server</td>
<td>cpu_count</td>
<td>/proc/cpuinfo</td>
</tr>
<tr>
<td>CPU core count</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_count</td>
<td>/proc/cpuinfo</td>
</tr>
<tr>
<td>Label</td>
<td>Table Name</td>
<td>Field Name</td>
<td>Source</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------</td>
<td>-------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>CPU core thread</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_thread</td>
<td>/proc/cpuinfo</td>
</tr>
<tr>
<td>CPU manufacturer</td>
<td>cmdb_ci_linux_server</td>
<td>cpu_manufacturer</td>
<td>/proc/cpuinfo</td>
</tr>
<tr>
<td>Model number</td>
<td>cmdb_ci_computer</td>
<td>model_number</td>
<td>dmidecode</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_computer</td>
<td>model_id</td>
<td>dmidecode</td>
</tr>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_linux_server</td>
<td>ram</td>
<td>meminfo</td>
</tr>
<tr>
<td>Disk space (GB)*</td>
<td>cmdb_ci_linux_server</td>
<td>disk_space</td>
<td>/proc/ide, /proc/scsi, /var/log/dmesg</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_ci_disk</td>
<td>type</td>
<td>/proc/ide, /proc/scsi, /var/log/dmesg</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_disk</td>
<td>model_id</td>
<td>/proc/ide, /proc/scsi, /var/log/dmesg</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_disk</td>
<td>disk_space</td>
<td>/proc/ide, /proc/scsi, /var/log/dmesg</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_disk</td>
<td>name</td>
<td>/proc/ide, /proc/scsi, /var/log/dmesg</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_file_system</td>
<td>name</td>
<td>df</td>
</tr>
<tr>
<td>Capacity (MB)</td>
<td>cmdb_ci_file_system</td>
<td>capacity</td>
<td>df</td>
</tr>
<tr>
<td>Available Space (MB)</td>
<td>cmdb_ci_file_system</td>
<td>available_space</td>
<td>df</td>
</tr>
<tr>
<td>Mount point</td>
<td>cmdb_ci_file_system</td>
<td>mount_point</td>
<td>df</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_running_process</td>
<td>name</td>
<td>ps</td>
</tr>
<tr>
<td>Command</td>
<td>cmdb_running_process</td>
<td>command</td>
<td>ps</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_running_process</td>
<td>type</td>
<td>ps</td>
</tr>
<tr>
<td>PID</td>
<td>cmdb_running_process</td>
<td>pid</td>
<td>ps</td>
</tr>
<tr>
<td>Parameters</td>
<td>cmdb_running_process</td>
<td>parameters</td>
<td>ps</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_network_adapter</td>
<td>name</td>
<td>ifconfig</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>ifconfig</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>ifconfig</td>
</tr>
<tr>
<td>Netmask</td>
<td>cmdb_ci_network_adapter</td>
<td>netmask</td>
<td>ifconfig</td>
</tr>
<tr>
<td>Default gateway</td>
<td>cmdb_ci_hardware</td>
<td>default_gateway</td>
<td>route</td>
</tr>
</tbody>
</table>

* The value in the **disk_space** field is an aggregation of the disk space for all non-removable disks, including both directly attached and SAN storage.

Discovery also identifies and classifies information about Linux KVM. Discovery identifies Linux kernel-based virtual machines (KVM) when the process classifier detects `libvirtd` running on a Linux server. The classification triggers the creation of a `cmdb_ci_kvm` record, and launches the SSHCommand probes to explore the Linux server with `virsh`, `lvbvert` utility, and virtual machine configuration data.

Discovery creates a `cmdb_ci_kvm_instance` record for each virtual machine on the server, and then matches the `cmdb_ci_kvm_instance` record to a corresponding `cmdb_ci_computer` record using the MAC addresses of installed network adapters.
### Tables used by Discovery on Linux KVM

<table>
<thead>
<tr>
<th>Table name</th>
<th>Extends</th>
<th>Description</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_kvm</td>
<td>cmdb_ci_vm</td>
<td>A hypervisor that manages kernel-based virtual machines (KVMs)</td>
<td>Process classifier detects libvirtd running on Linux servers</td>
</tr>
<tr>
<td>cmdb_ci_kvm_vm_instance</td>
<td>cmdb_ci_vm_instance</td>
<td>A virtual machine instance on this hypervisor</td>
<td>virsh list --all and dumpxml command</td>
</tr>
<tr>
<td>cmdb_ci_kvm_object</td>
<td>cmdb_ci_vm_object</td>
<td>An object connected to a virtual machine instance</td>
<td>&lt;network&gt;, &lt;storage pool&gt;, and &lt;storage volume&gt; elements from the dumpxml command</td>
</tr>
<tr>
<td>cmdb_kvm_device</td>
<td>N/A</td>
<td>A device connected to a virtual machine instance</td>
<td>&lt;devices&gt; element from the dumpxml command</td>
</tr>
</tbody>
</table>

### Data collected by Discovery on Linux KVM

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux Host</td>
<td>cmdb_ci_kvm</td>
<td>linux_host</td>
<td>Reference to the cmdb_ci_linux_server that is running this virtual machine</td>
</tr>
<tr>
<td>Details</td>
<td>cmdb_ci_kvm</td>
<td>details_xml</td>
<td>dumpxml</td>
</tr>
<tr>
<td>Object ID</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>object_id</td>
<td>virsh dumpxml</td>
</tr>
<tr>
<td>State</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>state</td>
<td>virsh list --all</td>
</tr>
<tr>
<td>CPUs</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>cpus</td>
<td>virsh dumpxml</td>
</tr>
<tr>
<td>Memory</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>memory</td>
<td>virsh dumpxml</td>
</tr>
<tr>
<td>Disks</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>disks</td>
<td>virsh dumpxml</td>
</tr>
<tr>
<td>Disks size</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>disks_size</td>
<td>virsh domblkinfo</td>
</tr>
<tr>
<td>Network adapters</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>nics</td>
<td>virsh dumpxml</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>name</td>
<td>virsh dumpxml</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_kvm_vm_instance</td>
<td>short_description</td>
<td>virsh desc</td>
</tr>
<tr>
<td>Details</td>
<td>cmdb_ci_kvm_object</td>
<td>details_xml</td>
<td>XML element from dumpxml</td>
</tr>
<tr>
<td>KVM instance</td>
<td>cmdb_kvm_device</td>
<td>kvm_instance</td>
<td>Reference to cmdb_ci_kvm_instance</td>
</tr>
<tr>
<td>Device</td>
<td>cmdb_kvm_device</td>
<td>device</td>
<td>disk, controller, interface, etc.</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_kvm_device</td>
<td>type</td>
<td>depends on device</td>
</tr>
<tr>
<td>Details</td>
<td>cmdb_kvm_device</td>
<td>details_xml</td>
<td>XML element from dumpxml</td>
</tr>
</tbody>
</table>
Data collected by Discovery on KVM Relationship

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Parent table</th>
<th>Child table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registered On::Has Registered</td>
<td>KVM (cmdb_ci_kvm)</td>
<td>KVM Virtual Machine Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(cmdb_ci_kvm_vm_instance)</td>
</tr>
<tr>
<td>Provided By::Provides</td>
<td>KVM (cmdb_ci_kvm)</td>
<td>Network</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(cmdb_ci_kvm_network)</td>
</tr>
<tr>
<td>Defines resource for::Gets resources from</td>
<td>KVM (cmdb_ci_kvm)</td>
<td>Storage Pool</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(cmdb_ci_kvm_storage_pool)</td>
</tr>
<tr>
<td>Connected By::Connects</td>
<td>KVM Virtual Machine Instance</td>
<td>Network</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_kvm_vm_instance)</td>
<td>(cmdb_ci_kvm_network)</td>
</tr>
<tr>
<td>Instantiated By::Instantiates</td>
<td>KVM Virtual Machine Instance</td>
<td>Computer</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_kvm_vm_instance)</td>
<td>(cmdb_ci_computer)</td>
</tr>
<tr>
<td>Virtualized By::Virtualizes</td>
<td>Computer (cmdb_ci_computer)</td>
<td>KVM (cmdb_ci_kvm)</td>
</tr>
<tr>
<td>Provides storage for::Stored on</td>
<td>Storage Pool (cmdb_ci_kvm_storage_pool)</td>
<td>KVM Virtual Machine Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(cmdb_ci_kvm_vm_instance)</td>
</tr>
</tbody>
</table>

Linux Red Hat cluster discovery
Discovery can find Linux Red Hat clusters that offer high availability and load balancing.

**Required credentials**

Configure [SSH credentials](#).

**Additional requirements**

You also need permissions to read the `/etc/hosts/` and `/etc/cluster/cluster.conf` file, which associates host names with IP addresses.

The following commands are run for Oracle clusterware:

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ps -ef</td>
</tr>
<tr>
<td>clustat -x</td>
</tr>
<tr>
<td>ifconfig</td>
</tr>
<tr>
<td>hostname -s</td>
</tr>
<tr>
<td>hostname -s</td>
</tr>
</tbody>
</table>

*Sudo permission to run:* sudo `/u01/app/12.1.0.2/grid/bin/ocrcheck | egrep -v 'error| return code'`
Classifier, probes, and pattern

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux (Unix classification)</td>
<td>Horizontal discovery probe: launches patterns</td>
<td>These patterns are an extension of the Linux Server pattern:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- UNIX Cluster - Linux Red hat cluster</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Linux - Red Hat Cluster</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The pattern uses the clustat -x command and looks at this cluster configuration file: /etc/cluster/cluster.conf</td>
</tr>
</tbody>
</table>

Data collected

<table>
<thead>
<tr>
<th>Table</th>
<th>Labels and field names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unix Cluster (cmdb_ci_unix_cluster)</td>
<td>Name (name)</td>
</tr>
<tr>
<td></td>
<td>Cluster status (cluster_status)</td>
</tr>
<tr>
<td></td>
<td>Cluster type (cluster_type)</td>
</tr>
<tr>
<td></td>
<td>IP address (ip_address)</td>
</tr>
<tr>
<td></td>
<td>Cluster ID (cluster_id)</td>
</tr>
<tr>
<td></td>
<td>Cluster version (cluster_version)</td>
</tr>
<tr>
<td>Unix Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>Name (name)</td>
</tr>
<tr>
<td></td>
<td>Node state (node_state)</td>
</tr>
<tr>
<td></td>
<td>IP address (ip_address)</td>
</tr>
<tr>
<td></td>
<td>Node status (node_status)</td>
</tr>
<tr>
<td>Unix Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
<td>Name (name)</td>
</tr>
<tr>
<td></td>
<td>Resource type (resource_type)</td>
</tr>
<tr>
<td></td>
<td>Resource status (resource_status)</td>
</tr>
<tr>
<td></td>
<td>Properties (properties)</td>
</tr>
<tr>
<td>Cluster Virtual IP (cmdb_ci_cluster_vip)</td>
<td>Name (name)</td>
</tr>
<tr>
<td></td>
<td>IP address (ip_address)</td>
</tr>
<tr>
<td></td>
<td>Cluster ID (cluster_id)</td>
</tr>
</tbody>
</table>

Relationships

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unix Cluster (cmdb_ci_unix_cluster)</td>
<td>Virtualizes::Virtualized by</td>
<td>Cluster Virtual IP (cmdb_ci_cluster_vip)</td>
</tr>
</tbody>
</table>
### Unix Cluster Resource (cmdb_ci_unix_cluster_resource) Relationship

** Defines resources for::Gets resources from ** Unix Cluster Node (cmdb_ci_unix_cluster_node)  

### Cluster Virtual IP (cmdb_ci_cluster_vip) Relationship

** Uses::Used by ** Unix Cluster Node (cmdb_ci_unix_cluster_node)  

### Unix Cluster Node (cmdb_ci_unix_cluster_node) Relationship

** Cluster of::Cluster ** Unix Cluster (cmdb_ci_unix_cluster)  

### Unix Cluster Node (cmdb_ci_unix_cluster_node) Relationship

** Hosted on::Hosts ** Unix Cluster (cmdb_ci_unix_cluster)  

### Unix Cluster Resource (cmdb_ci_unix_cluster_resource) Relationship

** Defines resources for::Gets resources from ** Unix Cluster (cmdb_ci_unix_cluster)  

### Unix Cluster (cmdb_ci_unix_cluster) Relationship

** Hosts::Hosted on ** Linux Server (cmdb_ci_linux_server)  

### Unix Cluster Node (cmdb_ci_unix_cluster_node) Relationship

** Hosted on::Hosts ** Linux Server (cmdb_ci_linux_server)  

### Netware discovery

Discovery identifies and classifies information about Netware.

#### Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ci_netware_server</td>
<td>name</td>
<td>snmp</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_netware_server</td>
<td>serial_number</td>
<td>snmp</td>
</tr>
<tr>
<td>OS Version</td>
<td>cmdb_ci_netware_server</td>
<td>os_version</td>
<td>snmp</td>
</tr>
<tr>
<td>RAM</td>
<td>cmdb_ci_netware_server</td>
<td>ram</td>
<td>snmp</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_netware_server</td>
<td>cpu_count</td>
<td>snmp</td>
</tr>
</tbody>
</table>

### OS/X (Mac) discovery

Discovery identifies and classifies information about Mac (OS/X) computers.

#### Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_computer</td>
<td>os</td>
<td>uname</td>
</tr>
<tr>
<td>OS Version</td>
<td>cmdb_ci_computer</td>
<td>os_version</td>
<td>system_profiler</td>
</tr>
<tr>
<td>OS Service pack</td>
<td>cmdb_ci_computer</td>
<td>os_service_pack</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_computer</td>
<td>short_description</td>
<td>uname</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_computer</td>
<td>name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_computer</td>
<td>host_name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_computer</td>
<td>dns_domain</td>
<td>DNS</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci_computer</td>
<td>start_date</td>
<td>uptime</td>
</tr>
<tr>
<td>Label</td>
<td>Table Name</td>
<td>Field Name</td>
<td>Source</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------</td>
<td>----------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_computer</td>
<td>manufacturer</td>
<td>Assumed to be Apple</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_computer</td>
<td>serial_number</td>
<td>system_profiler</td>
</tr>
<tr>
<td>CPU type</td>
<td>cmdb_ci_computer</td>
<td>cpu_type</td>
<td>system_profiler</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_computer</td>
<td>cpu_speed</td>
<td>system_profiler</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_computer</td>
<td>cpu_count</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_computer</td>
<td>model_id</td>
<td>system_profiler</td>
</tr>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_computer</td>
<td>ram</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_computer</td>
<td>disk_space</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Volume name</td>
<td>cmdb_ci_disk</td>
<td>volume_name</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Volume serial number</td>
<td>cmdb_ci_disk</td>
<td>volume_serial_number</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_disk</td>
<td>disk_space</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_disk</td>
<td>name</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Device ID</td>
<td>cmdb_ci_disk</td>
<td>device_id</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Free space (GB)</td>
<td>cmdb_ci_disk</td>
<td>free_space</td>
<td>system_profiler</td>
</tr>
<tr>
<td>File system</td>
<td>cmdb_ci_disk</td>
<td>file_system</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_running_process</td>
<td>name</td>
<td>ps</td>
</tr>
<tr>
<td>Command</td>
<td>cmdb_running_process</td>
<td>command</td>
<td>ps</td>
</tr>
<tr>
<td>Connects to</td>
<td>cmdb_running_process</td>
<td>connects_to</td>
<td>lsof</td>
</tr>
<tr>
<td>Listening on</td>
<td>cmdb_running_process</td>
<td>listening_on</td>
<td>lsof</td>
</tr>
<tr>
<td>PID</td>
<td>cmdb_running_process</td>
<td>pid</td>
<td>ps</td>
</tr>
<tr>
<td>Parameters</td>
<td>cmdb_running_process</td>
<td>parameters</td>
<td>ps</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_network_adapter</td>
<td>name</td>
<td>system_profiler</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>system_profiler</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>system_profiler</td>
</tr>
<tr>
<td>Netmask</td>
<td>cmdb_ci_network_adapter</td>
<td>netmask</td>
<td>system_profiler</td>
</tr>
<tr>
<td>MAC manufacturer</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_mfr</td>
<td>Assumed to be Apple</td>
</tr>
<tr>
<td>DHCP enabled</td>
<td>cmdb_ci_network_adapter</td>
<td>dhcp_enabled</td>
<td>system_profiler</td>
</tr>
</tbody>
</table>

**Solaris discovery**
Discovery identifies and classifies information about Solaris computers.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KBO694477](#) for more information.

**Supported operating systems**
- Oracle Solaris 10
- Oracle Solaris 11
Prerequisites

Discovery stores data in the (cmdb_running_process) table with truncated command line parameters up to 80 characters. This can cause multiple applications to be merged into one CI. To get the full command line and prevent this issue, run `pargs -a` and parse the result.

For Discovery to use patterns to find Solaris computers, you must install `SNEEP`. Otherwise, Discovery cannot find the serial number.

If using Solaris zones, each zone should be configured to return a unique value for the serial number. Otherwise, each zone will return the same number causing issues with CI identification.

Credentials

Configure SSH credentials for the systems you want to discover.

For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

Classifier, probes, and pattern

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
</table>
| Solaris (UNIX Classification) | • Horizontal discovery probe: launches patterns  
  • Solaris - Installed Software^  
  • Solaris - Identity*  
  • Solaris - CPU*  
  • Solaris - Hardware Model*  
  • Solaris - Memory*  
  • Solaris - Network ARP Tables*  
  • Solaris - OS Patches*  
  • Solaris - Storage*  
  • Solaris - Zones*  
  • Solaris - Disks*  
  • DNS* | Solaris Server |

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^This probe remains active by default, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_computer</td>
<td>os</td>
<td>uname</td>
</tr>
<tr>
<td>Label</td>
<td>Table name</td>
<td>Field name</td>
<td>Source</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------</td>
<td>-----------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_solaris_server</td>
<td>short_description</td>
<td>uname</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_solaris_server</td>
<td>name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_solaris_server</td>
<td>host_name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_solaris_server</td>
<td>dns_domain</td>
<td>DNS</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci_solaris_server</td>
<td>start_date</td>
<td>uptime</td>
</tr>
<tr>
<td>CPU type</td>
<td>cmdb_ci_computer</td>
<td>cpu_type</td>
<td>kstat</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_computer</td>
<td>cpu_speed</td>
<td>kstat</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_computer</td>
<td>cpu_count</td>
<td>kstat</td>
</tr>
<tr>
<td>CPU core count</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_count</td>
<td>kstat</td>
</tr>
<tr>
<td>CPU core thread</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_thread</td>
<td>kstat</td>
</tr>
<tr>
<td>Model number</td>
<td>cmdb_ci_solaris_server</td>
<td>model_number</td>
<td>suntype</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_solaris_server</td>
<td>model_id</td>
<td>suntype</td>
</tr>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_computer</td>
<td>ram</td>
<td>prtconf</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_solaris_server</td>
<td>disk_space</td>
<td>iostat</td>
</tr>
<tr>
<td>Serial Number</td>
<td>cmdb_ci_solaris_server</td>
<td>serial_number</td>
<td>sneep *</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_disk</td>
<td>serial_number</td>
<td>iostat</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_disk</td>
<td>model_id</td>
<td>iostat</td>
</tr>
<tr>
<td>Volume serial number</td>
<td>cmdb_ci_disk</td>
<td>volume_serial_number</td>
<td>iostat</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_disk</td>
<td>disk_space</td>
<td>iostat</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_disk</td>
<td>name</td>
<td>iostat</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_file_system</td>
<td>name</td>
<td>df</td>
</tr>
<tr>
<td>Capacity (MB)</td>
<td>cmdb_ci_file_system</td>
<td>capacity</td>
<td>df</td>
</tr>
<tr>
<td>Available Space (MB)</td>
<td>cmdb_ci_file_system</td>
<td>available_space</td>
<td>df</td>
</tr>
<tr>
<td>Mount point</td>
<td>cmdb_ci_file_system</td>
<td>mount_point</td>
<td>df</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_patches</td>
<td>name</td>
<td>showrev</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_running_process</td>
<td>name</td>
<td>ps</td>
</tr>
<tr>
<td>Command</td>
<td>cmdb_running_process</td>
<td>command</td>
<td>ps</td>
</tr>
<tr>
<td>Connects to</td>
<td>cmdb_running_process</td>
<td>connects_to</td>
<td>lsof</td>
</tr>
<tr>
<td>Listening on</td>
<td>cmdb_running_process</td>
<td>listening_on</td>
<td>lsof</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_running_process</td>
<td>type</td>
<td>ps</td>
</tr>
<tr>
<td>PID</td>
<td>cmdb_running_process</td>
<td>pid</td>
<td>ps</td>
</tr>
<tr>
<td>Parameters</td>
<td>cmdb_running_process</td>
<td>parameters</td>
<td>ps</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_network_adapter</td>
<td>name</td>
<td>ifconfig</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>ifconfig</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>ifconfig</td>
</tr>
<tr>
<td>Netmask</td>
<td>cmdb_ci_network_adapter</td>
<td>netmask</td>
<td>ifconfig</td>
</tr>
<tr>
<td>Label</td>
<td>Table name</td>
<td>Field name</td>
<td>Source</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------</td>
<td>---------------------</td>
<td>--------</td>
</tr>
<tr>
<td>Default gateway</td>
<td>cmdb_ci_hardware</td>
<td>default_gateway</td>
<td>netstat</td>
</tr>
</tbody>
</table>

* To discover Fujitsu PRIMEPOWER devices, you must install Oracle SNEEP and run Solaris discovery with root credentials.

**Solaris zone discovery**

Discovery maps the relationships between global and local Solaris zones upon detection.

In the following example, a Solaris global zone contains two local zones: zone01 and zone02. Each local zone is represented by a physical Solaris CI record and a Virtual Machine Instance record. Each of the local zones is tied to a Zone Server, demonstrating how virtualization relates to the global zone (mmp1).
The TCP connection and process information for local zone servers must be collected by running commands on their parent global zone. The relationship path between the local and global zone physical machines must be established before TCP connection and process information for local zone servers can be collected.

Case 1: Global zone discovered first.
- The system creates the Solaris server CI for the global zone.
- Discovery detects the local zones, creates a hypervisor zone server record, and creates a virtual machine instance record for each Solaris device in the local zone.
- Discovery creates the relationship between the hypervisor record and the VM instance record.

Case 2: Local zone discovered first.
- The system creates the Solaris server CI for the local zone.
- Discovery sets the correlation ID, so that it can be reconciled during later global zone discoveries.

Case 3: Global zone discovered after creation of local zone Solaris server CIs.
- Global zone Discovery detects local zones.
- Discovery creates a hypervisor zone server record, and creates a virtual machine instance record for each Solaris device in the local zone.
- Discovery creates the relationship between the hypervisor record and the VM instance record.
  In addition, it creates the relationship between the physical local zone VM and its virtual machine instance record.
- The global zone runs the Solaris - ADM probe on itself, filtering by the local zone, and updates the physical local zone VMs with this data.

Case 4: The relationship path between physical local and global zone machines is established.

Subsequent discoveries of the global zone refresh the TCP connection and process information for the contained local zones.

When the system discovers a global zone, the Solaris - Zones & ADM Launcher probe triggers the Solaris - ADM probe to explore the global zone and each local zone found. Because the Solaris - ADM probe must run on the global zone to detect TCP connection and process information from its local zones, you might see multiple ECC queue records that appear identical.
ECC queue entries for a zone Discovery

Upon examining the payload, however, you will see that each probe is actually targeting a different zone CI to filter on and update.

```xml
<parameter name="sequence" value="1523a01c9470000001"/>
<parameter name="from host" value=""/>
<parameter name="zone" value="zone01"/>
<parameter name="sys_created_on" value="2016-01-13 02:02:29"/>
<parameter name="used_by_discovery" value="true"/>
```

Local zone payload

**Data collected on Solaris zones**

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>cmdb_ci_vm_zones</td>
<td>version</td>
<td>zoneadm, zonename</td>
</tr>
<tr>
<td>Correlation ID</td>
<td>cmdb_ci_vm_zones</td>
<td>correlation_id</td>
<td>zoneadm, zonename</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_vm_instance</td>
<td>name</td>
<td>zoneadm, zonename</td>
</tr>
<tr>
<td>Parent</td>
<td>cmdb_ci_vm_instance</td>
<td>parent</td>
<td>Internal</td>
</tr>
<tr>
<td>CMDB CI</td>
<td>cmdb_ci_vm_instance</td>
<td>cmdb_ci</td>
<td>Internal</td>
</tr>
<tr>
<td>Correlation ID</td>
<td>cmdb_ci_vm_instance</td>
<td>correlation_id</td>
<td>zoneadm, zonename</td>
</tr>
</tbody>
</table>

**Virtual machine (vm) discovery**

Discovery identifies and classifies information about virtual machines.
### VMware discovery

A Discovery schedule can discover VMware vCenter and ESX hosts.

These options are available for getting VMware vCenter data:

- Discovery runs the **VMware - vCenter Datacenters** probe when it identifies a VMware vCenter process running on a Windows machine or detects activity with the vmapp port probe.
- **Orchestration** can run any of the vCenter probes from a workflow.

#### Discovery for VMware vCenter

Discovery can explore the VMware vCenter process running on a Windows or Linux host.

### Tested vCenter versions

The ServiceNow® platform supports vCenter API versions 4.0 and higher. The following versions were tested with the New York release:

- vCenter versions 6.7 and earlier
- vSphere version 5.5
- vCenter appliance version 6.7 and earlier

**Note:** If you are discovering SUSE Linux hosts for vCenter appliances, version 6.0 and earlier, observe these **SSH restrictions**.

See [Data collected for VMware vCenter Server](#) for a description of the VMware architecture and component relationships.

### vCenter discovery process

After classifying vCenter, Discovery launches the VMware - vCenter Datacenters probe, which in turn launches specific probes that return information about ESX machines, virtual machines, and other vCenter objects. The vmapp port probe is also configured to launch the VMware - vCenter Datacenters probe.
VMware credentials

If you use a domain account to access vCenter, specify the domain with the user name in the credential record in one of the supported formats, such as Domain\UserName.

Attention: Windows credentials are not necessary for vCenter Discovery, when valid VMware credentials are used.

Note: The VMware credentials must have the read-only role and License Admin privilege in vCenter.

Updating the CMDB with vCenter event collector

In addition to finding vCenter data through the standard discovery process, Discovery can also update the CMDB by detecting vCenter events through a MID Server extension called the vCenter event collector.

The event collector allows the CMDB to be updated with changes to virtual machines (VMs), in addition to the updates detected by Discovery. A change to a VM is sent as an event from the vCenter server to the vCenter event collector. When an event is received, the CMDB is updated accordingly. Full vCenter Discovery does not need to rerun. For some events, such as powered on and powered off events, Discovery does not need to run again at all. For most events, Discovery runs only on the necessary vCenter resource.

For instructions on configuring vCenter events, see Configure and run the vCenter event collector extension.

CIs removed from vCenter

When a vCenter CI, such as a virtual machine, is removed, the ServiceNow instance marks it as “stale” in the CMDB, using either of these procedures:

- When Discovery runs, it creates an audit record in the CMDB Health Result (cmdb_health_result) table for the missing CI and marks the CI “stale”.
- If the instance is configured to collect vCenter events, the system can also create a “stale” audit record for the CI in the CMDB Health Result (cmdb_health_result) table from the VmRemovedEvent event, without having to run Discovery.

Note: When the Staleness setting is configured, the dependency view (BSM map) grays out stale CIs in its relationship diagram to indicate that they were removed from vCenter.
vCenter CI marked as “stale”

You have the option of creating a CMDB remediation rule to automatically execute a remediation workflow that can, for example, delete stale CIs. For more information on stale CIs, see CMDB Health Metrics.

vCenter Discovery on Windows host

Windows credentials are not necessary for vCenter Discovery, when valid VMware credentials are used.

Configure an alternate port for vCenter

You can specify an alternate port for the VMWare - vCenter Datacenters probe.

Role required: admin

By default, the VMWare - vCenter Datacenters probe runs on port 443, which is the standard port for the https protocol. The port probes for vCenter run on these ports:

- **vmapp6_https**: 9443
• vmapp_https: 5480

1. Hard code the port information.
   a) Navigate to Configuration > VMware > vCenter.
   b) Select the specific instance from the list of instances.
   c) Edit the URL field to include the port.
      For example, https://10.0.0.1:444
   d) Click Submit.

2. Specify an alternate port number for vCenter in the VMWare - vCenter Datacenters probe.
   a) Navigate to Discovery > Probes.
   b) Open the VMware - vCenter Datacenters probe record.
   c) In the Probe Parameters related list, click New.
   d) In the Name field of the Probe parameter record, enter vcenter_port.
   e) In the Value field, enter your alternate port number.
   f) Click Submit.

Data collected for VMware vCenter Server
Discovery identifies and classifies information about VMware vCenter servers.

vCenter table schema

The vCenter table schema is illustrated in the following diagram:
Several tables are cloud-agnostic tables, meaning that they can be populated for any cloud resource, not just vCenter resources. Look in the sub tables that extend the cloud-agnostic tables to find discovered CIs.

<table>
<thead>
<tr>
<th>Cloud-agnostic table</th>
<th>Look in this vCenter-specific table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Machine Instance (cmdb_ci_vm_instance)</td>
<td>VMware Virtual Machine Instance (cmdb_ci_vmware_instance)</td>
</tr>
<tr>
<td>Operating System Template (cmdb_ci_os_template)</td>
<td>VMware Virtual Machine Template (cmdb_ci_vmware_template)</td>
</tr>
<tr>
<td>Logical Datacenter (cmdb_ci_logical_datacenter)</td>
<td>VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Cloud Networks (cmdb_ci_network)</td>
<td>VMware vCenter Network (cmdb_ci_vcenter_network)</td>
</tr>
</tbody>
</table>
Cloud-agnostic table | Look in this vCenter-specific table
--- | ---
Datastore (cmdb_ci_datastore) | VMware vCenter Datastore (cmdb_ci_vcenter_datastore)
Host Cluster (cmdb_ci_host_cluster) | VMware vCenter Cluster (cmdb_ci_vcenter_cluster)

vCenter data

Discovery uses multiple vCenter probes to collect this data from vCenter. The data is saved in tables extend from the Configuration item (cmdb_ci) table.

**VMware vCenter Instance (cmdb_ci_vcenter)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Full name</td>
<td>fullname</td>
</tr>
<tr>
<td>Instance UUID</td>
<td>instance_uuid</td>
</tr>
<tr>
<td>URL</td>
<td>url</td>
</tr>
<tr>
<td>Effective CPU</td>
<td>effectivecpu</td>
</tr>
</tbody>
</table>

The combination of the vCenter instance in the cmdb_ci_vcenter table and the object ID in the cmdb_ci_vm_object table identifies a specific vCenter.

The tables extend from the Virtual Machine Objects (cmdb_ci_vm_object) table, which holds object IDs for all cloud-based resources, except virtualization servers.

**DRS VM Config (cmdb_ci_drs_vm_config)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>cluster</td>
</tr>
<tr>
<td>DRS behavior</td>
<td>drs_behavior</td>
</tr>
<tr>
<td>DRS enabled</td>
<td>drs_enabled</td>
</tr>
<tr>
<td>Virtual machine</td>
<td>virtual_machine</td>
</tr>
</tbody>
</table>

This table stores DRS behavior for the VMs that override the cluster behavior. For more information on DRS, see VMware's documentation on Distributed Resource Scheduler.

**VMware Virtual Machine Instance (cmdb_ci_vmware_instance)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Template</td>
<td>template</td>
</tr>
<tr>
<td>CPUs</td>
<td>cpus</td>
</tr>
<tr>
<td>Disks</td>
<td>disks</td>
</tr>
<tr>
<td>Disks size (GB)</td>
<td>disks_size</td>
</tr>
</tbody>
</table>
**Field label | Column name**
--- | ---
Memory (MB) | memory
Network adapters | nics
Image path | image_path
State | state
vCenter Instance UUID | vcenter_uuid
vCenter Reference | vcenter_ref
Base name | base_name
VM namer | vm_namer
BIOS UUID | bios_uuid
VM Instance UUID | vm_instance_uuid

**VMware Distributed Virtual Port Groups (cmdb_ci_vcenter_dv_port_group)**

| Field Label | Column name |
--- | --- |
Name | name
vCenter Instance UUID | vcenter_uuid
Distributed Virtual Switch Reference | dvs_ref

**VMware Virtual Machine Template (cmdb_ci_vmware_template)**

| Field label | Column name |
--- | --- |
Name | The template name from VMWare.
CPUs | cpus
Disks size (GB) | disks_size
Guest ID | guest_id
Memory (MB) | memory
Image path | image_path
State | state
vCenter Instance UUID | vcenter_uuid
vCenter Reference | vcenter_ref
Network adapters | nics
Base name | base_name
VM namer | vm_namer
BIOS UUID | bios_uuid
VM Instance UUID | vm_instance_uuid
**VMware vCenter Object (cmdb_ci_vcenter_object)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
</tbody>
</table>

**VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region</td>
<td>region</td>
</tr>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
<tr>
<td>Top level folder for VMs</td>
<td>folder_moreid</td>
</tr>
<tr>
<td>Top level folder for hosts</td>
<td>host_morid</td>
</tr>
</tbody>
</table>

*The VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter) table is extended from the (cmdb_ci_logical_datacenter) table.*

**VMware vCenter Network (cmdb_ci_vcenter_network)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
<tr>
<td>Network accessible by either hosts or virtual machines</td>
<td>accessible</td>
</tr>
</tbody>
</table>

*The VMware vCenter Network (cmdb_ci_vcenter_network) table is extended from the (cmdb_ci_network) table.*

**VMware Distributed Virtual Switch (cmdb_ci_vcenter_dvs)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMCount</td>
<td>vm_count</td>
</tr>
<tr>
<td>Host count</td>
<td>host_count</td>
</tr>
</tbody>
</table>

**VMware vCenter Folder (cmdb_ci_vcenter_folder)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
</tbody>
</table>
### ESX Resource Pool (cmdb_ci_esx_resource_pool)

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
<tr>
<td>Owner</td>
<td>owner</td>
</tr>
<tr>
<td>Owner Managed Object Reference ID</td>
<td>owner_morid</td>
</tr>
<tr>
<td>CPU expandable</td>
<td>cpu_expandable</td>
</tr>
<tr>
<td>CPU limit (MHz)</td>
<td>cpu_limit_mhz</td>
</tr>
<tr>
<td>CPU reserved (MHz)</td>
<td>cpu_reserved_mhz</td>
</tr>
<tr>
<td>CPU shares</td>
<td>cpu_shares</td>
</tr>
<tr>
<td>Full path</td>
<td>fullpath</td>
</tr>
<tr>
<td>Memory expandable</td>
<td>mem_expandable</td>
</tr>
<tr>
<td>Memory limit (MB)</td>
<td>mem_limit_mb</td>
</tr>
<tr>
<td>Memory reserved (MB)</td>
<td>mem_reserved_mb</td>
</tr>
<tr>
<td>Memory shares</td>
<td>mem_shares</td>
</tr>
</tbody>
</table>

### VMware vCenter Datastore (cmdb_ci_vcenter_datastore)

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>Accessible</td>
<td>accessible</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
<tr>
<td>Capacity (GB)</td>
<td>capacity</td>
</tr>
<tr>
<td>Free space (GB)</td>
<td>freespace</td>
</tr>
<tr>
<td>URL</td>
<td>url</td>
</tr>
<tr>
<td>Clustered</td>
<td>clustered</td>
</tr>
</tbody>
</table>

The VMware vCenter Datastore (cmdb_ci_vcenter_datastore) table is extended from the (cmdb_ci_datastore) table.

### VMware vCenter Cluster (cmdb_ci_vcenter_cluster)*

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managed object reference ID</td>
<td>morid</td>
</tr>
<tr>
<td>Field label</td>
<td>Column name</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>Effective hosts</td>
<td>effectivehosts</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
<tr>
<td>Effective CPU</td>
<td>effectivecpu</td>
</tr>
<tr>
<td>Effective memory</td>
<td>effectivememory</td>
</tr>
<tr>
<td>Number of effective hosts</td>
<td>effectivehosts</td>
</tr>
<tr>
<td>Number of hosts</td>
<td>numhosts</td>
</tr>
<tr>
<td>Total CPU</td>
<td>totalcpu</td>
</tr>
<tr>
<td>Total memory</td>
<td>totalmemory</td>
</tr>
<tr>
<td>Number of CPU cores</td>
<td>numcpucores</td>
</tr>
<tr>
<td>Number of CPU threads</td>
<td>numcputhreads</td>
</tr>
<tr>
<td>DRS Behavior</td>
<td>drs_behavior</td>
</tr>
<tr>
<td>DRS Enabled</td>
<td>drs_enabled</td>
</tr>
<tr>
<td>DRS VMotion Rate</td>
<td>drs_vmotion_rate</td>
</tr>
</tbody>
</table>

*The VMware vCenter Cluster (cmdb_ci_vcenter_cluster) table is extended from the (cmdb_ci_host_cluster) table.

**VMware Distributed Virtual Switch (cmdb_ci_vcenter_dvs)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>VMCount</td>
<td>vm_count</td>
</tr>
<tr>
<td>Host count</td>
<td>host_count</td>
</tr>
</tbody>
</table>

Discovery also maps the relationships to VMs and to distributed virtual port groups.

**VMware Distributed Virtual Port Group (cmdb_ci_vcenter_dv_port_group)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>vCenter Instance UUID</td>
<td>vcenter_uuid</td>
</tr>
<tr>
<td>Distributed Virtual Switch Reference</td>
<td>dvs_ref</td>
</tr>
</tbody>
</table>

Discovery also maps the relationship to the distributed virtual switch.

You can add these related lists to view additional discovered data:

- **Storage Volumes**
  
The virtual disks for this virtual machine. This data is saved in the Storage Volume (cmdb_ci_storage_volume) table with the value type=v Disk.
Storage Volume (cmdb_ci_storage_volume)

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Size</td>
<td>size</td>
</tr>
</tbody>
</table>

- **Network Adapters**

  The virtual network adapters for the virtual disks. This data is saved in the Network Adapter (cmdb_ci_network_adapter) table.

  **Network Adapter (cmdb_ci_network_adapter)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>MAC address</td>
<td>mac_address</td>
</tr>
<tr>
<td>Netmask</td>
<td>netmask</td>
</tr>
</tbody>
</table>

**vCenter discovery with Software Asset Management**

If Software Asset Management is active, Discovery populates these vCenter tables using the VMWare - vCenter ESX Hosts License probe.

**VMWare License Key (samp_vmware_license_key)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>vCenter Reference</td>
<td>vcenter_ref</td>
</tr>
<tr>
<td>Cost Unit</td>
<td>cost_unit</td>
</tr>
<tr>
<td>Edition</td>
<td>edition</td>
</tr>
<tr>
<td>Features</td>
<td>features</td>
</tr>
<tr>
<td>License Key</td>
<td>license_key</td>
</tr>
<tr>
<td>Product Name</td>
<td>product_name</td>
</tr>
<tr>
<td>Product Version</td>
<td>product_version</td>
</tr>
<tr>
<td>Rights Owned</td>
<td>rights Owned</td>
</tr>
<tr>
<td>Rights Used</td>
<td>rights_used</td>
</tr>
</tbody>
</table>

**VMWare License Key Usage (samp_vmware_license_key_usage)**

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rights Used</td>
<td>rights_used</td>
</tr>
<tr>
<td>Expiration Date</td>
<td>expiration_date</td>
</tr>
<tr>
<td>Used Features</td>
<td>used_features</td>
</tr>
</tbody>
</table>
### vCenter relationships

Discovery automatically creates relationships for vCenter components using data from a key class. Subsequent discoveries use the same key class to automatically validate and remove relationships that are no longer valid.

vCenter CIs can be members of folders or clusters, which affect how Discovery creates their relationships.

- If a CI is in a folder, Discovery creates a relationship between that CI and the folder. If that CI is not in a folder, Discovery creates the relationship between the CI and the datacenter. These vCenter CIs can be in a folder:
  - VM Instance
  - VM Template
  - vCenter Network
  - Datastore
  - vCenter Folder
  - vCenter Cluster

- If an ESX server is in a cluster, Discovery creates a relationship between the ESX server and the cluster. If an ESX server is not a member of a cluster, then Discovery creates a relationship to the datacenter.

- If a resource pool is in a cluster, Discovery creates a relationship between the resource pool and the cluster. If the resource pool is not a member of a cluster, then Discovery creates a relationship to the ESX server.

This diagram illustrates vCenter relationships:

<table>
<thead>
<tr>
<th>Field label</th>
<th>Column name</th>
</tr>
</thead>
<tbody>
<tr>
<td>License Key</td>
<td>license_key</td>
</tr>
<tr>
<td>Software Install</td>
<td>software_install</td>
</tr>
<tr>
<td>Used By</td>
<td>used_by</td>
</tr>
</tbody>
</table>
vCenter Instance
cmdb_ci_vcenter

Manages::Managed by

Contains::Contained by

Contains::Contained by

Contains::Contains

Datacenter
cmdb_ci_vcenter_datacenter

Provides storage for::Stores

Contains::Contained by

Cluster
cmdb_ci_vcenter_cluster

Gets resources from::Resources for

Members::Member of

Resource Pools
cmdb_ci_esx_resource_pool

Contains::Contained by

Provides::Provides

ESXi Server
cmb_ci_esx_server

Has registered::Registered

Has registered::Registered

Uses::Used by

rtu

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Parent class</th>
<th>Relationship type</th>
<th>Child class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer (cmdb_ci_computer)</td>
<td>Virtualized by::Virtualizes</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>Computer (cmdb_ci_computer)</td>
<td>Instantiated by::Instantiated by</td>
<td>VM Instance (cmdb_ci_vmware_instance)</td>
</tr>
<tr>
<td>VMware Virtual Machine Instance (cmdb_ci_vmware_instance)</td>
<td>Registered on::Has registered</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>VMware Virtual Machine Instance (cmdb_ci_vmware_instance)</td>
<td>Connected by::Connects</td>
<td>VMware vCenter Network (cmdb_ci_vcenter_network)</td>
</tr>
<tr>
<td>Virtual Machine Template (cmdb_ci_vmware_template)</td>
<td>Connected by::Connects</td>
<td>VMware vCenter Network (cmdb_ci_vcenter_network)</td>
</tr>
<tr>
<td>VMware vCenter Network (cmdb_ci_vcenter_network)</td>
<td>Provided by::Provides</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>Provides storage for::Stored on</td>
<td>VMware Virtual Machine Instance (cmdb_ci_vmware_instance)</td>
</tr>
<tr>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>Used by::Uses</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>Provides storage for::Stored on</td>
<td>Virtual Machine Template (cmdb_ci_vmware_template)</td>
</tr>
<tr>
<td>VMware vCenter Cluster (cmdb_ci_vcenter_cluster)</td>
<td>Members::Member of</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>Defines resources for::Get resources from</td>
<td>VMware vCenter Cluster (cmdb_ci_vcenter_cluster)</td>
</tr>
<tr>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>Defines resources for::Get resources from</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>VMware vCenter Folder (cmdb_ci_vcenter_folder)</td>
<td>Contains::Contained by</td>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
</tr>
<tr>
<td>VMware vCenter Folder (cmdb_ci_vcenter_folder)</td>
<td>Contains::Contained by</td>
<td>VMware vCenter Folder (cmdb_ci_vcenter_folder)</td>
</tr>
<tr>
<td>VMware vCenter Folder (cmdb_ci_vcenter_folder)</td>
<td>Contains::Contained by</td>
<td>Virtual Machine Template (cmdb_ci_vmware_template)</td>
</tr>
<tr>
<td>VMware vCenter Folder (cmdb_ci_vcenter_folder)</td>
<td>Contains::Contained by</td>
<td>VMware Virtual Machine Instance (cmdb_ci_vmware_instance)</td>
</tr>
<tr>
<td>VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
<td>Contains::Contained by</td>
<td>VMware vCenter Network (cmdb_ci_vcenter_network)</td>
</tr>
<tr>
<td>VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
<td>Contains::Contained by</td>
<td>VMware Virtual Machine Instance (cmdb_ci_vmware_instance)</td>
</tr>
<tr>
<td>VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
<td>Contains::Contained by</td>
<td>ESX Server (cmdb_ci_esx_server)</td>
</tr>
<tr>
<td>VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
<td>Contains::Contained by</td>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
</tr>
<tr>
<td>VMware vCenter Datacenter (cmdb_ci_vcenter_datacenter)</td>
<td>Contains::Contained by</td>
<td>VMware vCenter Folder (cmdb_ci_vcenter_folder)</td>
</tr>
</tbody>
</table>
Cloud Management relationships

These additional relationships are created when Cloud Management (CMP) is active.

<table>
<thead>
<tr>
<th>Parent class</th>
<th>Relationship type</th>
<th>Child class</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware vCenter Datacenter</td>
<td>Contains::Contained by</td>
<td>VMware vCenter Cluster</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_datacenter)</td>
<td></td>
<td>(cmdb_ci_vcenter_cluster)</td>
</tr>
<tr>
<td>VMware vCenter Datacenter</td>
<td>Contains::Contained by</td>
<td>Virtual Machine Template</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_datacenter)</td>
<td></td>
<td>(cmdb_ci_vmware_template)</td>
</tr>
<tr>
<td>ESX Server</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_esx_server)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>vCenter Folder</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_folder)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>vCenter Datacenter</td>
<td>Hosted on::Hosts</td>
<td>Cloud Service Account</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_datacenter)</td>
<td></td>
<td>(cmdb_ci_cloud_service_account)</td>
</tr>
<tr>
<td>vCenter Cluster</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_cluster)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Resource Pools</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_esx_resource_pool)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>VM Instance</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vmware_instance)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>VM Template</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vmware_template)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>vCenter Network</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_network)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Distributed Virtual Switch</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_dvs)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Distributed Virtual Port Group</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_dv_port_group)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Datastore</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vcenter_datastore)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Virtual Disk</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_storage_volume)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>Virtual NIC</td>
<td>Hosted on::Hosts</td>
<td>vCenter Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_vmware_nic)</td>
<td></td>
<td>(cmdb_ci_vcenter_datacenter)</td>
</tr>
<tr>
<td>VM Template</td>
<td>Use End Point To::Use End</td>
<td>Block Endpoint</td>
</tr>
<tr>
<td>(cmdb_ci_vmware_template)</td>
<td>Point From</td>
<td>(cmdb_ci_endpoint_block)</td>
</tr>
<tr>
<td>VM Instance</td>
<td>Use End Point To::Use End</td>
<td>Block Endpoint</td>
</tr>
<tr>
<td>(cmdb_ci_vmware_instance)</td>
<td>Point From</td>
<td>(cmdb_ci_endpoint_block)</td>
</tr>
<tr>
<td>Virtual Disk</td>
<td>Implement End Point To::</td>
<td>Block Endpoint</td>
</tr>
<tr>
<td>(cmdb_ci_storage_volume)</td>
<td>Implement End Point From</td>
<td>(cmdb_ci_endpoint_block)</td>
</tr>
</tbody>
</table>
VMware tags

You can attach tags to vSphere objects, such as virtual machines, through the vSphere interface. The tags can then be grouped into categories. Objects with tags are sortable and searchable based on the parameters you give the tags and categories. The VMWare — vCenter VM Tags probe discovers these tags. You can view the discovered tags by opening the virtual machine record in `cmdb_ci_vmware_instance`. Then go to the Key Values tab.

**Key Values (cmdb_key_value)**

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration item (configuration_item)</td>
<td>Referenced VM with tag attached in vCenter.</td>
</tr>
<tr>
<td>Key (key)</td>
<td>Category name of the tag applied to the resource in vCenter.</td>
</tr>
<tr>
<td>Tag (tag)</td>
<td>Source in vCenter from which tags are fetched. In this case, the value is always Tags.</td>
</tr>
<tr>
<td>Value (value)</td>
<td>Tag name that is applied to the resource in vCenter.</td>
</tr>
</tbody>
</table>

Datastore data collected by Discovery

Discovery identifies each datastore in the system and creates the relationships with the virtual machines and the ESX servers that use these datastores.

**Datastores**

Discovery uses the VMWare - vCenter Datastores probe to collect this data from datastores.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Table Name</th>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessible</td>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>accessible</td>
<td>Whether the datastore is collected or not.</td>
</tr>
<tr>
<td>Capacity (GB)</td>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>capacity</td>
<td>Amount of space provided by the datastore.</td>
</tr>
<tr>
<td>Clustered</td>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>clustered</td>
<td>If the datastore is clustered (belongs to a storage pod).</td>
</tr>
<tr>
<td>Free space (GB)</td>
<td>VMware vCenter Datastore (cmdb_ci_vcenter_datastore)</td>
<td>freespace</td>
<td>Amount of space still available on the datastore.</td>
</tr>
</tbody>
</table>
Field Label | Table Name | Column Name | Description
--- | --- | --- | ---
Type | VMware vCenter Datastore (cmdb_ci_vcenter_datastore) | type | The type of file system volume, such as WFS or NFS.
URL | VMware vCenter Datastore (cmdb_ci_vcenter_datastore) | url | The unique URL locator for the datastore.

HostMounts

Discovery uses both the VMWare - vCenter ESX Hosts and VMWare - vCenter Datastores probes to collect datastore host mount data.

<table>
<thead>
<tr>
<th>Field label</th>
<th>Table</th>
<th>Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessible</td>
<td>VMware Datastore HostMount (vcenter_datastore_hostmount)</td>
<td>accessible</td>
</tr>
<tr>
<td>Access Mode</td>
<td>VMware Datastore HostMount (vcenter_datastore_hostmount)</td>
<td>access_mode</td>
</tr>
<tr>
<td>VMware vCenter Datastore</td>
<td>VMware Datastore HostMount (vcenter_datastore_hostmount)</td>
<td>datastore</td>
</tr>
<tr>
<td>ESX Server</td>
<td>VMware Datastore HostMount (vcenter_datastore_hostmount)</td>
<td>esx_server</td>
</tr>
<tr>
<td>vCenter Reference</td>
<td>VMware Datastore HostMount (vcenter_datastore_hostmount)</td>
<td>vcenter_ref</td>
</tr>
</tbody>
</table>

ESX server discovery

Discovery identifies and classifies information about ESX servers and ESX resource pools through the discovery of vCenter and not from the direct discovery of any ESX servers.

**Important:** ESX server discovery is done through vCenter. Do not specify the IP address of the ESX server in a Discovery Schedule. Instead, discover the vCenter through the Discovery Schedule.

For a description of the VMware architecture and component relationships, see [Data collected for VMware vCenter Server](#).

For a discussion of how Discovery collects information on datastores and establishes the relationships between the ESX hosts and the storage disks attached to the datastores, see [Datastore Discovery](#).

**Required Roles**

Users with the `itil` and `asset` roles can access ESXi and ESXi configuration item (CI) records. To run discovery on vCenter servers, users must have the `discovery_admin` role.
Credentials

To run a complete Discovery of vCenter/ESX servers, you need vCenter credentials. If the vmapp port probe is disabled, you must use Windows credentials to access the Windows host on which the vCenter server runs.

**Note:** Make sure to select a credential Type of VMware.

ESX server Discovery components

Discovery identifies ESX servers based on the combination of correlation ID (BIOS UUID) and Managed Object Reference ID, when the hardware manufacturer is on a certified white list. Manufacturers not on this list are identified by serial number. After running the vCenter classifier, Discovery launches the VMware - vCenter Datacenters probe, which launches the probes that explore the ESX server. For the complete list of vCenter probes, see [List of Discovery probes](#).

<table>
<thead>
<tr>
<th>Component</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classifier</td>
<td>vCenter</td>
<td>Classifies stand-alone vCenter servers.</td>
</tr>
<tr>
<td>Probe</td>
<td>VMWare - vCenter ESX Hosts</td>
<td>Creates records for ESX servers and hostmounts. Creates relationships between ESX servers and vCenter components. Triggers probes for storage Discovery.</td>
</tr>
<tr>
<td>Probe</td>
<td>VMWare - vCenter ESX Hosts Storage</td>
<td>Creates records for ESX host hardware: network adapters, disks, HBAs, FC ports, iSCSI and FC disks. Creates relationships between DAS/iSCSI/FC disks and datastore disks.</td>
</tr>
</tbody>
</table>

Data collected

Basic server data from ESX hosts is collected by the VMware - vCenter ESX Hosts probe.

**Data collected by Discovery on ESX hosts**

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_esx_server</td>
<td>os</td>
</tr>
<tr>
<td>OS Version</td>
<td>cmdb_ci_computer</td>
<td>os_version</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_esx_server</td>
<td>short_description</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_esx_server</td>
<td>name</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_esx_server</td>
<td>dns_domain</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci_esx_server</td>
<td>start_date</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_computer</td>
<td>manufacturer</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_computer</td>
<td>serial_number</td>
</tr>
<tr>
<td>Label</td>
<td>Table Name</td>
<td>Field Name</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>CPU type</td>
<td>cmdb_ci_esx_server</td>
<td>cpu_type</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_esx_server</td>
<td>cpu_speed</td>
</tr>
<tr>
<td>CPU count</td>
<td>cmdb_ci_esx_server</td>
<td>cpu_count</td>
</tr>
<tr>
<td>CPU core count</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_count</td>
</tr>
<tr>
<td>CPU core thread</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_thread</td>
</tr>
<tr>
<td>CPU manufacturer</td>
<td>cmdb_ci_esx_server</td>
<td>cpu_manufacturer</td>
</tr>
<tr>
<td>Model number</td>
<td>cmdb_ci_computer</td>
<td>model_number</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_computer</td>
<td>model_id</td>
</tr>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_esx_server</td>
<td>ram</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_esx_server</td>
<td>disk_space</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_ci_disk</td>
<td>type</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_disk</td>
<td>model_id</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_disk</td>
<td>disk_space</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_disk</td>
<td>name</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_network_adapter</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
</tr>
<tr>
<td>Netmask</td>
<td>cmdb_ci_network_adapter</td>
<td>netmask</td>
</tr>
<tr>
<td>Default gateway</td>
<td>cmdb_ci_hardware</td>
<td>default_gateway</td>
</tr>
<tr>
<td>Managed object reference ID</td>
<td>Visualization Server (cmdb_ci_virtualization_server)</td>
<td>morid</td>
</tr>
<tr>
<td>Serial Number</td>
<td>Serial Number (cmdb_serial_number)</td>
<td>serial_number</td>
</tr>
</tbody>
</table>

**Relationships**

Discovery collects the following relationship data for ESX servers.

<table>
<thead>
<tr>
<th>Base Class</th>
<th>Relationship</th>
<th>Dependent Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>Defines resources for: Gets resources from ESX Server (cmdb_ci_esx_server)</td>
<td></td>
</tr>
<tr>
<td>VM Instance (cmdb_ci_vmware_instance)</td>
<td>Registered on: Has registered ESX Server (cmdb_ci_esx_server)</td>
<td></td>
</tr>
<tr>
<td>ESX Server (cmdb_ci_esx_server)</td>
<td>Provides storage for: Stored on VM Template (cmdb_ci_vmware_template)</td>
<td></td>
</tr>
<tr>
<td>ESX Server (cmdb_ci_esx_server)</td>
<td>Provided by: Provides Networks (cmdb_ci_vcenter_network)</td>
<td></td>
</tr>
<tr>
<td>ESX Server (cmdb_ci_esx_server)</td>
<td>Members of: Members Cluster (cmdb_ci_vcenter_cluster)</td>
<td></td>
</tr>
</tbody>
</table>
## ESX resource pools

Resource pools are configured in vCenter and define the maximum amount of resources that virtual machines using that pool can consume. An ESX server property enables resource pools to expand when necessary if the ESX server has additional resources to spare. The **Name** and **Owner** fields of each resource pool on the ESX server must be configured within the Now Platform in the ESX Resource Pool (cmdb_ci_esx_resource_pool) table. When Orchestration for VMware executes its manual provisioning tasks, the provisioner must select the proper resource pool for the virtual server requested. Discovery finds resource pools on ESX machines and populates the fields on the ESX Resource Pool form automatically. For more information, see [Configure ESX resource pools](#). ESX resource pools requires the Orchestration - VMware Support plugin.

**Note:** Ensure that vCenter and the ESX server have been fully configured, including the creation of the templates and resource pools.

### Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU expandable</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>cpu_expendable</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>CPU limit (MHz)</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>cpu_limit_mhz</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>CPU reserved (MHz)</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>cpu_reserved_mhz</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>CPU shares</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>cpu_shares</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Full path</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>fullpath</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Memory expandable</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>mem_expansible</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Memory limit (MB)</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>mem_limit_mb</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Memory reserved (MB)</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>mem_reserved_mb</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Memory shares</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>mem_shares</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Owner</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>owner</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
<tr>
<td>Owner Managed</td>
<td>ESX Resource Pool (cmdb_ci_esx_resource_pool)</td>
<td>owner_morid</td>
<td>VMWare - vCenter Clusters probe</td>
</tr>
</tbody>
</table>

### Configure ESX resource pools
The ESX server has a default resource pool called Resources that defines normal resources for a virtual machine.

Role required: admin

Resource levels are dynamically generated from shares of the total resources allocated to virtual machines on the ESX server. For details about how these resources are calculated, review the VMware Knowledge Base https://www.vmware.com/. ServiceNow Discovery finds this default resource pool and adds a record to the ESX Resource Pools module automatically. If Discovery is not running on the ServiceNow instance, create a record for the Resources pool. Ensure that the Owner field is correct and leave the resource fields blank. If a provisioner selects the Resources pool when provisioning a virtual server, the ESX server will create a virtual machine for use under a normal load.

2. Click New in the list.
3. Create a new record for each resource pool in the ESX server, ensuring that the Name and Owner fields are correct.
4. Select the CPU expandable and Memory expandable check boxes to allow for expansion of the CPU and memory limits when needed, if those resources are available on the ESX server.

When granted, these extra resources can be revoked if needed to provision other virtual machines. The additional fields on the form are for display purposes only.
5. Click Submit.
ServiceNow New York IT Operations Management

ESX Resource Pool Resources

Name: Resources
Owner: DISCO-CLUSTER1

CPU reserved (MHz): 252396
CPU limit (MHz): 193396
CPU shares: 4080
CPU expandable: ✓

Memory reserved (MB): 1018780
Memory limit (MB): 1018780
Memory shares: 163840
Memory expandable: ✓

Related Items

Maximum relationship limit has been reached and not all relationships were shown in theformatter.

Downstream relationships

- [Members] 104_CentOS64
- [Members] 2012Win-MWD
- [Members] 2008HyperV
- [Members] 2nd Low Disk Space - CentOS
- [Members] SCPU_2Cores
- [Members] 24_Win2008R2_1E1_NET4.6.2_64-bit
- [Members] 381.0.11.128.238 NOServer Win2008R2_1E1_NET4.5_64-bit
- [Members] Admin Test: Linux1
- [Members] Admin Test: Linux2
- [Members] Admin Test: Solaris

Upstream relationships

- [Resources] Resources
VMware virtual machines

Discovery gathers information about virtual machines managed by vCenter.

The Discovery product can extract information from VMware configuration items (CI), including their relationships. Discovery is configured to work with the Orchestration VMware Support Plugin, which enables the automatic cloning of virtual machines by an ESX server managed by vCenter. For information about data collected by Discovery on vCenter, see Data collected for VMware vCenter Server.

VMware Workstation - Legacy

In the basic VMware system, the VMware Workstation runs on a Windows or Linux host machine.

**Attention:** The Discovery of VMware Workstation is not supported in releases after Kingston. The VMware workstation probe and sensor are not included in new instances of London. However, customers who upgrade from a release prior to Kingston can continue to use the probe and sensor to discover VMware Workstation.

This system can clone instances from templates, but cannot be automated. The relationships between VMware components for this type of installation are shown in the following diagram:

![VMware Workstation relationships](image)

**VMware Workstation component relationships**

<table>
<thead>
<tr>
<th>Component</th>
<th>Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows or Linux Server</td>
<td>Runs the VMware application</td>
</tr>
<tr>
<td>VMware application</td>
<td>- Runs on a Windows or Linux host machine</td>
</tr>
<tr>
<td></td>
<td>- Has registered VM instances</td>
</tr>
<tr>
<td></td>
<td>- Virtualizes virtual machines</td>
</tr>
<tr>
<td>VM Instances (including images and templates)</td>
<td>- Registers on the VMware application</td>
</tr>
<tr>
<td></td>
<td>- Instantiated by individual virtual machines</td>
</tr>
<tr>
<td>Virtual machines</td>
<td>- Instantiates VM instances</td>
</tr>
<tr>
<td></td>
<td>- Virtualized by VMware application</td>
</tr>
</tbody>
</table>

Datastore Discovery
A datastore is a storage object for virtual machines that are hosted on an ESX server. A datastore is a collection of one or more physical disks, such as an iSCSI disk, and can be used by more than one ESX host. However, a physical disk can only connect to one datastore. Because ESX hosts can share datastores, it is easy to move virtual machine between hosts that have a common datastore.

**Note:** From the perspective of a virtual machine attached to a datastore, storage is provided by a single disk.

The advantages to connecting a datastore to multiple disks are:
- The ability to mirror the disks for failover protection.
- The ease of adding storage to the datastore.

For information about the tables used by Discovery to store the data for physical disks and their relationship to datastores and ESX hosts, see [ESX server discovery](#).

In this example configuration, two ESX hosts share a common datastore that uses different types of storage.
Relationships

ServiceNow provides tables that contain the relationships between an ESX host and its datastore and the specific disks to which the datastore is connected.

ServiceNow Discovery establishes the relationships between a datastore, the disks attached to the datastore, and the ESX server that hosts the virtual machines using that datastore. From the perspective of the ESX host, iSCSI and fibre channel disks connected to the datastore are treated as physical disks. Discovery does not show the direct relationship of the storage disks to the ESX host.

Note: Storage can be hosted on computers that are not discovered by Discovery. ESX hosts are discovered through vCenter, and storage is discovered separately through CIM. The system can only establish the relationship between the two when storage is discovered before ESX.

Datastore tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_vcenter_datastore_disk</td>
<td>Stores the relationship of the physical disk to the datastore.</td>
</tr>
<tr>
<td>vcenter_datastore_hostmount</td>
<td>Stores the relationship between the datastore and the ESX server to which it is connected.</td>
</tr>
<tr>
<td>cmdb_ci_disk</td>
<td>Contains the physical disks connected directly to the datastore. This table also contains a reference to the ESX host.</td>
</tr>
<tr>
<td>cmdb_ci_fc_disk</td>
<td>Contains the fibre channel disks in a storage area network (SAN) connected to the datastore. This table also contains a reference to the ESX host.</td>
</tr>
<tr>
<td>cmdb_ci_iscsi_disk</td>
<td>Contains the iSCSI disks in an IP network connected to the datastore. This table also contains a reference to the ESX host.</td>
</tr>
</tbody>
</table>

Windows discovery

Discovery identifies and classifies information about Windows computers.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Supported Windows versions

- Windows Workstation 7, 8, 10
- Windows Server versions:
  - 2008
  - 2008R2
  - 2012R2
  - 2016
  - 2019
**Note:** For fiber channel discovery on a Windows 2008 host, the Microsoft Fibre Channel Information Tool (fcinfo.exe) must be installed on that machine. The fcinfo executable should be available on the environment path. The Microsoft Fibre Channel Information Tool is available for download at [http://www.microsoft.com](http://www.microsoft.com).

**Credentials**

Configure **Windows credentials**. The user configured in the credentials must have local admin access to the Windows machine.

The credential that you configure must have the following:

- Access to the WMI service to the current namespace and sub-namespaces.
- Access to the Powershell service.
- Membership in the Distributed COM Users local security group.

**Classifiers, probes, and patterns**

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Probes</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>Horizontal discovery probe: launches patterns</td>
<td>Windows OS - Server</td>
</tr>
<tr>
<td>Windows 2008 Server</td>
<td>WMIRunner-Windows - Installed Software^</td>
<td>Windows OS - Desktops</td>
</tr>
<tr>
<td>Windows 2012 Server</td>
<td>MultiProbe-Windows - ADMA</td>
<td></td>
</tr>
<tr>
<td>Windows 2016 Server</td>
<td>Windows - Identity*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Network ARP Table*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - OS Information*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Cluster*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - CPU/Memory*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Installed Software*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Printers*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Storage 2008*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Storage 2012*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Amazon EC2*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows - Azure*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DNS*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SNMP - Routing*</td>
<td></td>
</tr>
</tbody>
</table>

Windows 2019 Server   Windows OS - Server

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^These probes remain active by default, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.
## Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assigned to</td>
<td>cmdb_ci_win_server</td>
<td>assigned_to</td>
<td>wmi</td>
</tr>
<tr>
<td>Chassis type</td>
<td>cmdb_ci_win_server</td>
<td>chassis_type</td>
<td>wmi</td>
</tr>
<tr>
<td>Command</td>
<td>cmdb_running_process</td>
<td>command</td>
<td>wmi</td>
</tr>
<tr>
<td>Connects to</td>
<td>cmdb_running_process</td>
<td>connects_to</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU core count*</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_count</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU core thread*</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_thread</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU count*</td>
<td>cmdb_ci_computer</td>
<td>cpu_count</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU manufacturer</td>
<td>cmdb_ci_computer</td>
<td>cpu_manufacturer</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU name</td>
<td>cmdb_ci_computer</td>
<td>cpu_name</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_computer</td>
<td>cpu_speed</td>
<td>wmi</td>
</tr>
<tr>
<td>Default gateway</td>
<td>cmdb_ci_win_server</td>
<td>default_gateway</td>
<td>wmi</td>
</tr>
<tr>
<td>Department</td>
<td>cmdb_ci_win_server</td>
<td>department</td>
<td>Internal (User)</td>
</tr>
<tr>
<td>Description</td>
<td>cmdb_ci_disk</td>
<td>short_description</td>
<td>wmi</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_computer</td>
<td>disk_space</td>
<td>wmi</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_disk</td>
<td>disk_space</td>
<td>wmi</td>
</tr>
<tr>
<td>DHCP enabled</td>
<td>cmdb_ci_network_adapter</td>
<td>dhcp_enabled</td>
<td>wmi</td>
</tr>
<tr>
<td>DNS domain</td>
<td>cmdb_ci_win_server</td>
<td>dns_domain</td>
<td>DNS</td>
</tr>
<tr>
<td>Free space (GB)</td>
<td>cmdb_ci_file_system</td>
<td>free_space</td>
<td>wmi</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_win_server</td>
<td>host_name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>wmi</td>
</tr>
<tr>
<td>Listening on</td>
<td>cmdb_running_process</td>
<td>listening_on</td>
<td>wmi</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>wmi</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_win_server</td>
<td>manufacturer</td>
<td>wmi</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci</td>
<td>model_id</td>
<td>wmi</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_win_server</td>
<td>name</td>
<td>DNS, NBT</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_disk</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_running_process</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_network_adapter</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>Netmask</td>
<td>cmdb_ci_network_adapter</td>
<td>netmask</td>
<td>wmi</td>
</tr>
<tr>
<td>Operating System</td>
<td>cmdb_ci_computer</td>
<td>os</td>
<td>wmi</td>
</tr>
<tr>
<td>OS domain</td>
<td>cmdb_ci_computer</td>
<td>os_domain</td>
<td>NBT</td>
</tr>
<tr>
<td>OS service pack</td>
<td>cmdb_ci_computer</td>
<td>os_service_pack</td>
<td>wmi</td>
</tr>
<tr>
<td>OS version</td>
<td>cmdb_ci_computer</td>
<td>os_version</td>
<td>wmi</td>
</tr>
<tr>
<td>Parameters</td>
<td>cmdb_running_process</td>
<td>parameters</td>
<td>wmi</td>
</tr>
<tr>
<td>PID</td>
<td>cmdb_running_process</td>
<td>pid</td>
<td>wmi</td>
</tr>
</tbody>
</table>
### Table

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_computer</td>
<td>ram</td>
<td>wmi</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_win_server</td>
<td>serial_number</td>
<td>wmi</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_win_server</td>
<td>short_description</td>
<td>wmi</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_ci_disk</td>
<td>type</td>
<td>wmi</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_running_process</td>
<td>type</td>
<td>wmi</td>
</tr>
<tr>
<td>Vendor</td>
<td>cmdb_ci_network_adapter</td>
<td>vendor</td>
<td>wmi</td>
</tr>
<tr>
<td>Volume serial number</td>
<td>cmdb_ci_disk</td>
<td>volume_serial_number</td>
<td>wmi</td>
</tr>
</tbody>
</table>

* Core counts and threads per core might not be accurate, due to issues with Microsoft reporting.

** The value in the **disk_space** field is an aggregation of the disk space for all non-removable disks, including both directly attached and SAN storage.

### The Windows registry

Discovery can find software that has been installed on a Windows machine by looking at the Windows Registry. Discovery can find the following attributes of discovered software:

- **Product Name**: Combination of name and version, such as Windows Imaging Component 3.0.
- **Name**: Name of the product only without the version.
- **Version**: Version of the product.
- **Uninstall String**: Path to the uninstaller, such as `C:\Program Files\Notepad++\uninstall.exe`.
- **Part of**: Update for which this is a part, such as Windows Internet Explorer 8 - Software U.
- **Install Date**: Date the software was installed. The Windows - Installed Software sensor appends a timestamp of 00:00:00 to the install_date retrieved from the registry. The installation time of all Windows software is independent of the timezone and is set to midnight of the day it was installed. For example, an install date of 2.19.2017 in the Windows registry appears as 2.19.2017 00:00:00 in the CMDB.
- **Installed on**: Name of the asset on which the software is installed.

### Windows server cluster discovery

Discovery establishes the relationships between a Windows server cluster and its nodes. When a Windows server is found, and Discovery detects that it is part of a cluster, the system launches the Windows - Cluster probe. Navigate to **Discovery Definition > Probes** and open the record for the Windows - Cluster probe to see the specific data Discovery collects.

*Note*: Discovery collects cluster resources based on the MSCluster_ResourceType WMI class. To see a list of default cluster resources, see **Resource Types**. To understand how cluster resources are related to process classifiers, see **Relate the process classifier to Windows cluster resources**.

### Cluster data tables

Discovery creates CIs for clusters in these tables:

- **Windows Cluster (cmdb_ci_win_cluster)**
- **Windows Cluster Resource(s) (cmdb_ci_win_cluster_resource)**
Cluster relationships

Discovery creates a CI Relationship (cmdb_rel_ci) record for each node, using these relationships:

- **Cluster of::Cluster**: Relationship between the cluster nodes and the cluster. Service Mapping uses this relationship to map the Windows cluster and its nodes.
- **Members::Member of**: Relationship between the cluster and the cluster node.
- **Hosted on::Hosts**: Relationship between the cluster nodes and the Windows server.

Windows cluster map

Windows cluster node relationships

Cluster references

Discovery creates the following cluster references:

- From Windows Cluster Nodes to the associated Windows Cluster.
- From Windows Cluster Resources to the associated Windows Cluster.
- From Cluster Virtual IPs to the associated Windows Cluster.
- From Cluster Node Resources to the associated Windows Cluster Node.

**Note:** References are maintained in Discovery to provide backward compatibility for systems that use them.

Help the Help Desk discovery on Windows
Help the Help Desk identifies and classifies information about Windows computers.

**Supported Windows Operating Systems**

- Windows 2008 and 2012 Server
- Windows XP and Vista
- Windows 7, 8, and 10

**Data collected**

Help the Help Desk stores information about Windows computers in the following tables and fields.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>cmdb_ci_computer</td>
<td>os</td>
<td>wmi</td>
</tr>
<tr>
<td>OS version</td>
<td>cmdb_ci_computer</td>
<td>os_version</td>
<td>wmi</td>
</tr>
<tr>
<td>OS service pack</td>
<td>cmdb_ci_computer</td>
<td>os_service_pack</td>
<td>wmi</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_computer</td>
<td>name</td>
<td>ActiveX</td>
</tr>
<tr>
<td>Hostname</td>
<td>cmdb_ci_computer</td>
<td>host_name</td>
<td>ActiveX</td>
</tr>
<tr>
<td>OS domain</td>
<td>cmdb_ci_computer</td>
<td>os_domain</td>
<td>wmi</td>
</tr>
<tr>
<td>Assigned to</td>
<td>cmdb_ci_computer</td>
<td>assigned_to</td>
<td>wmi</td>
</tr>
<tr>
<td>Department</td>
<td>cmdb_ci_computer</td>
<td>department</td>
<td>Internal (User)</td>
</tr>
<tr>
<td>Short description</td>
<td>cmdb_ci_computer</td>
<td>short_description</td>
<td>wmi</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci_computer</td>
<td>manufacturer</td>
<td>wmi</td>
</tr>
<tr>
<td>Serial number</td>
<td>cmdb_ci_computer</td>
<td>serial_number</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU name</td>
<td>cmdb_ci_computer</td>
<td>cpu_name</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU manufacturer</td>
<td>cmdb_ci_computer</td>
<td>cpu_manufacturer</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU speed (MHz)</td>
<td>cmdb_ci_computer</td>
<td>cpu_speed</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU count *</td>
<td>cmdb_ci_computer</td>
<td>cpu_count</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU core count *</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_count</td>
<td>wmi</td>
</tr>
<tr>
<td>CPU core thread *</td>
<td>cmdb_ci_computer</td>
<td>cpu_core_thread</td>
<td>wmi</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci_computer</td>
<td>model_id</td>
<td>wmi</td>
</tr>
<tr>
<td>RAM (MB)</td>
<td>cmdb_ci_computer</td>
<td>ram</td>
<td>wmi</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_computer</td>
<td>disk_space</td>
<td>wmi</td>
</tr>
<tr>
<td>Field Label</td>
<td>Table Name</td>
<td>Field Name</td>
<td>Source</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------</td>
<td>-------------------</td>
<td>--------</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_ci_disk</td>
<td>type</td>
<td>wmi</td>
</tr>
<tr>
<td>Description</td>
<td>cmdb_ci_disk</td>
<td>short_description</td>
<td>wmi</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>cmdb_ci_disk</td>
<td>disk_space</td>
<td>wmi</td>
</tr>
<tr>
<td>Free space (GB)</td>
<td>cmdb_ci_disk</td>
<td>free_space</td>
<td>wmi</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_disk</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>Volume serial number</td>
<td>cmdb_ci_disk</td>
<td>volume_serial_number</td>
<td>wmi</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_network_adapter</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>wmi</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>wmi</td>
</tr>
<tr>
<td>Netmask</td>
<td>cmdb_ci_network_adapter</td>
<td>netmask</td>
<td>wmi</td>
</tr>
<tr>
<td>DHCP enabled</td>
<td>cmdb_ci_network_adapter</td>
<td>dhcp_enabled</td>
<td>wmi</td>
</tr>
<tr>
<td>Vendor</td>
<td>cmdb_ci_network_adapter</td>
<td>vendor</td>
<td>wmi</td>
</tr>
<tr>
<td>Default gateway</td>
<td></td>
<td>default_gateway</td>
<td>wmi</td>
</tr>
</tbody>
</table>


**z/OS discovery**
Discovery finds computers running the z/OS operating system.

ℹ️ **Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](https://www.servicenow.com/knowledgearticle/KB0694477) for more information.

**Requirements**

The USS service must be running on port 22 on z/OS.

These commands may be run on the host machine:

- `uname`
- `hostname`
- `sysvar SYSNAME`
- `netstat`
- `df`
- `ls`
- `cat`
- `grep`
- `ps`
- `cut`

**Credentials**

Create [SSH credentials](https://www.servicenow.com/knowledgearticle/KB0694477). The user must have permission to write to `/tmp`, have read access to all user processes, and be able to run REXX scripts.
Classifier, probe, and pattern

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probe</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM z/OS</td>
<td>Horizontal Pattern: launches patterns</td>
<td>IBM zOS Server</td>
</tr>
</tbody>
</table>

Data collected

Discovery populates the IBM zOS server (cmdb_ci_ibm_zos_server) table with the following information:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>OS Version</td>
<td>os_version</td>
</tr>
<tr>
<td>Operating system</td>
<td>os</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>manufacturer</td>
</tr>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
</tbody>
</table>

Network device discovery

Discovery identifies several network devices, including load balancers, routers, TCP connections, IP networks, and so on. It can find devices through traditional port probes, and then through identifications and exploration probes, or through patterns. It also supports discovery by identifying HTTP and TCP connections.

Load balancer discovery

Discovery can collect data about network routers, switches, and applications.

Discovery supports data collection from the following applications and hardware:

- A10
- Apache mod_jk and Apache mod_proxy
- Big-IP F5 Traffic Manager
- Cisco
- Citrix NetScaler
- HAProxy
- NGINX
- Alteon
- ACE
- Radware

**Important:** Discovery treats hardware load balancers as network devices and attempts to discover them primarily using SNMP. If a load balancer in your system, running on a Linux host, has SNMP and SSH ports open, Discovery might classify it based on the SSH port. This classification has priority over SNMP. To ensure that Discovery properly classifies your hardware load balancers, create a Discovery behavior for load balancers that includes SNMP but not SSH. Software load balancers are treated as applications.
Load balancer patterns

Starting with the Jakarta release, Discovery uses out-of-box patterns to find some types load balancers for new instances (see table).

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

<table>
<thead>
<tr>
<th>Load balancer</th>
<th>Pattern names and description</th>
<th>Main CI table populated</th>
</tr>
</thead>
<tbody>
<tr>
<td>A10</td>
<td>A10 LoadBalancer</td>
<td>A10 Load Balancer (cmdb_ci_lb_a10)</td>
</tr>
<tr>
<td>ACE</td>
<td>ACE Load Balancer by SSH</td>
<td>ACE (cmdb_ci_lb_ace)</td>
</tr>
<tr>
<td>Alteon</td>
<td>Alteon Load Balancer</td>
<td>Alteon (cmdb_ci_lb_alteon)</td>
</tr>
<tr>
<td>F5 Big-IP</td>
<td>F5 - HD, F5 Load Balancer</td>
<td>F5 BIG-IPs (cmdb_ci_lb_bigip)</td>
</tr>
<tr>
<td>NetScaler</td>
<td>Netscaler Load Balancer</td>
<td>Citrix NetScalers (cmdb_ci_lb_netscaler)</td>
</tr>
<tr>
<td>Radware - AppDirector</td>
<td>AppDirector Load Balancer</td>
<td>Radware Load Balancer (cmdb_ci_lb_radware)</td>
</tr>
</tbody>
</table>

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Load balancer discovery, Service Mapping, and customized CI fields

Discovery collects information about the install status and operational status of load balancers. If you are using Discovery with Service Mapping, do not customize the (operational_status) or (install_status) fields. By default, Service Mapping ignores all host CIs for which the value of the (operational_status) field is not 1 (Operational) or the value of the status (install_status) field is 100 (absent). For additional information, see Preparing customized ServiceNow deployments to work with Service Mapping (KB0647574) in the HI Knowledge Base.

Load Balancer: A10
Discovery identifies and classifies information about A.10 load balancers.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Prerequisites and important information

Important: Discovery treats hardware load balancers as network devices and attempts to discover them primarily using SNMP. If a load balancer in your system, running on a Linux host, has SNMP and SSH ports open, Discovery might classify it based on the SSH port. This classification has priority over SNMP. To ensure that Discovery properly classifies your hardware load balancers, create a Discovery behavior for load balancers that includes SNMP but not SSH. Software load balancers are treated as applications.
## Credentials

Configure **SNMP credentials** or (optionally) **SSH credentials**.

## Classifier, pattern, and probe

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>A10 Load Balancer</td>
<td>• Horizontal discovery probe: launches patterns</td>
<td>A10 Load Balancer</td>
</tr>
</tbody>
</table>

## Data collected

The following data is collected on the A10 Load Balancers [cmdb_ci_lb_a10] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number</td>
<td>serial_number</td>
</tr>
<tr>
<td>Operational Status</td>
<td>operational_status</td>
</tr>
<tr>
<td>Label</td>
<td>Field Name</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Port</td>
<td>port</td>
</tr>
<tr>
<td>Operational Status</td>
<td>operational_status</td>
</tr>
<tr>
<td>Install Status</td>
<td>install_status</td>
</tr>
</tbody>
</table>

The following data is collected on the DNS [cmdb_ci_dns_name] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
</tbody>
</table>

**Relationships**

In this example from the dependency view, an A10 load balancer is represented by these elements:

- **Vip1**: A load balancer service that is the entry point for all business services attempting to contact the load balancer.
- **vthunder**: Actual A10 load balancer CI created by Discovery.
- **webApp-group1**: A load balancer pool containing the web server pool members IBM web01 and IBM web02.
A10 load balancer dependency view (BSM map)

Discovery creates these relationships from the perspective of the load balancer group:

A10 load balancer relationships

ServiceWatch maps an A10 load balancer like this:
A10 load balancer ServiceWatch map

For the top-down discovery of an A10 load balancer, relationships between the application and the load balance service are created only if the application CI exists.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

**Load balancer: ACE**

Discovery identifies and classifies information about ACE load balancers.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Credentials**

Configure [SNMP community credentials](#) or [SSH credentials](#).

**Classifier, pattern, and probe**

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE Load Balancer</td>
<td>Horizontal discovery probe: launches patterns</td>
<td>ACE Load Balancer by SSH (formerly ACE Load Balancer)</td>
</tr>
</tbody>
</table>

**Data collected**

The following data is collected on the Ace Load Balancers [cmdb_ci_Lb_Ace] table.
The following data is collected on the Load Balancer Services (cmdb_ci_lb_service) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number</td>
<td>serial_number</td>
</tr>
<tr>
<td>Operational Status</td>
<td>operational_status</td>
</tr>
</tbody>
</table>

The following data is collected on the DNS (cmdb_ci_dns_name) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
</tbody>
</table>

The following data is collected on the Load Balancer Pool (cmdb_ci_lb_pool) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Context</td>
<td>context</td>
</tr>
<tr>
<td>Load balancing method</td>
<td>load_balancing_method</td>
</tr>
</tbody>
</table>

The following data is collected on the Load Balancer Pool Members (cmdb_ci_lb_pool_member) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Service port</td>
<td>service_port</td>
</tr>
<tr>
<td>Pool</td>
<td>pool</td>
</tr>
<tr>
<td>Context</td>
<td>context</td>
</tr>
</tbody>
</table>

The following data is collected on the Serial Number (cmdb_serial_number) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
<tr>
<td>Serial number type</td>
<td>serial_number_type</td>
</tr>
</tbody>
</table>
The following data is collected on the Load Balancer Interface table (cmdb_ci_lb_interface) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>valid</td>
</tr>
</tbody>
</table>

Load balancer: Alteon
Discovery identifies and classifies information about Alteon load balancers.

- **Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Credentials

Configure [SNMP credentials](#).

### Classifier, pattern, and probe

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probe</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alteon Load Balancer</td>
<td>Horizontal pattern</td>
<td>Alteon Load Balancer by SNMP (formerly Alteon Load Balancer)</td>
</tr>
</tbody>
</table>

### Data collected

The following data is collected on the Alteon [cmdb_ci_lb_alteon] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number</td>
<td>serial_number</td>
</tr>
<tr>
<td>Operational Status</td>
<td>operational_status</td>
</tr>
</tbody>
</table>

The following data is collected on the Load Balancer Interface table (cmdb_ci_lb_interface) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>nam</td>
</tr>
<tr>
<td>MAC address</td>
<td>mac_address</td>
</tr>
<tr>
<td>Operational status</td>
<td>operational_status</td>
</tr>
</tbody>
</table>
The following data is collected on the Load Balancer Pool (cmdb_ci_lb_pool) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Index</td>
<td>index</td>
</tr>
<tr>
<td>Load balancing method</td>
<td>load_balancing_method</td>
</tr>
</tbody>
</table>

The following data is collected on the Load Balancer Pool Member (cmdb_ci_pool_member) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Service port</td>
<td>service_port</td>
</tr>
</tbody>
</table>

The following data is collected on the Load Balancer Services (cmdb_ci_lb_service) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Port</td>
<td>port</td>
</tr>
<tr>
<td>Operational status</td>
<td>operational_status</td>
</tr>
<tr>
<td>Install status</td>
<td>install_status</td>
</tr>
<tr>
<td>Hit count</td>
<td>hit_count</td>
</tr>
</tbody>
</table>

The following data is collected on the DNS (cmdb_ci_dns_name) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
</tbody>
</table>

Load balancer: Cisco GSS

Discovery of Cisco GSS load balancers is performed by both SNMP and SSH.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Classifiers**

Discovery uses the Cisco GSS Load Balancer classifier, which contains the condition: `sysdescr contains Global Site Selector` to classify the device.
Probes

Discovery uses several probes, including an SNMP probe to discover the specific GSS device and a Serial Number probe to identify the serial number of the device. See the table below for the list of related probes.

⚠️ Warning: You must have SNMP and SSH credentials configured correctly for the GSS load balancers on your network.

The following probes are triggered after classification:

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco GSS - Identity</td>
<td>A multiprobe that identifies the GSS device and serial number.</td>
<td>MultiProbe</td>
</tr>
<tr>
<td>Cisco GSS - Network</td>
<td>This is included in the Cisco GSS - Identify probe.</td>
<td>Java- SNMP</td>
</tr>
<tr>
<td>Cisco GSS - Serial Number</td>
<td>This is included in the Cisco GSS - Identify probe.</td>
<td>JavaScript</td>
</tr>
<tr>
<td>Cisco GSS - Get Domains</td>
<td>A probe that discovers global domain names corresponding to the device.</td>
<td>JavaScript</td>
</tr>
</tbody>
</table>

Tables

Discovery creates a record for each GSS device in the Cisco GSS (cmdb_ci_lb_cisco_gss) table. Domains are populated in the DNS Names (cmdb_ci_dns_name) table. Host names and IP addresses are stored in the IP Address to DNS Name (cmdb_ip_address_dns_name) table. Service Mapping uses this information. You can see the DNS name information on the DNS Names for CIs related list of the Load Balancer form.

ℹ️ Note: The Cisco GSS pattern is available by default to use with the Pattern Designer.

Load balancer: Cisco CSS

Discovery of Cisco CSS load balancers is performed by SNMP.

ℹ️ Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Classifiers

Discovery uses the Cisco CSS Load Balancer classifier, which contains the OID Classification: 1.3.6.141.9.9.368.4:

Probes

The following probes are triggered:
<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP - Load Balancer - Identity</td>
<td>A multiprobe that identifies load balancers.</td>
<td>Java-SNMP</td>
</tr>
<tr>
<td>Cisco CSS - Get Services</td>
<td>A Java probe that includes the Cisco CSS sensor to identify services defined on the load balancer in the Load Balancer Services (cmdb_ci_lb_service) table. For every service, Discovery populates Name, ip_address and port.</td>
<td>Java-SNMP</td>
</tr>
</tbody>
</table>

**Tables**

Discovery creates a record for each CSS device in the Cisco CSS (cmdb_ci_lb_cisco_css) table, which includes the device's Name, Model, Serial Number, Manufacture, and NIC (ip_address). It also creates a record for each service in the Load Balancer Services (cmdb_ci_lb_service) table.

Load balancer: Citrix NetScaler

Discovery and Service Mapping find Citrix NetScaler load balancers including Server Load Balancing (GSLB).

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Requirements and support**

**Important:** Discovery treats hardware load balancers as network devices and attempts to discover them primarily using SNMP. If a load balancer in your system, running on a Linux host, has SNMP and SSH ports open, Discovery might classify it based on the SSH port. This classification has priority over SNMP. To ensure that Discovery properly classifies your hardware load balancers, create a Discovery behavior for load balancers that includes SNMP but not SSH. Software load balancers are treated as applications.

Virtual NetScaler load balancers are not fully supported. Discovery supports local server load balancers and GSLB.

**Credentials**

- Configure [SNMP credentials](#).

  **Note:** You also have the option of using [SSH credentials](#). However, if the payload size of the data that is returned to the instance is large, the discovery process can fail. In this case, use SNMP credentials instead.

- (Optional) Provide a user with the read access to the NS.conf configuration file if you do not want to provide SNMP or SSH credentials.

- Use [basic authentication](#) credentials for discovery via REST.
For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

Classifier, pattern, and probes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetScaler Load Balancer</td>
<td>• Horizontal discovery probe: launches patterns (see the Patterns column)</td>
<td>• NetScaler Load Balancer</td>
</tr>
<tr>
<td></td>
<td>• NetScaler - Network Path Information (multiprobe)</td>
<td>• NetScaler Load Balancer NS.CONF</td>
</tr>
<tr>
<td></td>
<td>• NetScaler - ARP Information</td>
<td>• NetScaler Load Balancer SSH</td>
</tr>
<tr>
<td></td>
<td>• NetScaler - Route Information</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SNMP - NetScaler - Identity</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SNMP - NetScaler - Identity - Serial</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SNMP - NetScaler - Identity Info</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SNMP - NetScaler - System</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[cmdb_ci_lb_netscaler]</td>
<td></td>
</tr>
</tbody>
</table>

Horizontal discovery probe: launches patterns (see the Patterns column)

NetScaler - Network Path Information (multiprobe)

NetScaler - ARP Information

NetScaler - Route Information

SNMP - NetScaler - Identity

SNMP - NetScaler - Identity - Serial

SNMP - NetScaler - Identity Info

SNMP - NetScaler - System

SNMP - NetScaler - Identity Info

SNMP - NetScaler - Identity - Serial

SNMP - NetScaler - System

To use probes for the horizontal discovery, make sure that the NetScaler - Network Path Information multiprobe is active on the Trigger probes related list for the NetScaler Load Balancer classifier. Then deactivate or remove the Horizontal discovery probe from the Trigger probes related list.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Connections discovered by Service Mapping during the top-down discovery

Service Mapping performs the top-down discovery of the NetScaler Global Server Load Balancer in the context of application services. It discovers configured URLs of the NetScaler Global Server Load Balancer.

Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS name (cmdb_ci_dns_name)</td>
<td>The name or the DNS name alias of the NetScaler Global Server Load Balancing (GSLB).</td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>IP address (ip_addresses)</td>
<td>Management IP address of the NetScaler Global Server Load Balancing (GSLB).</td>
</tr>
<tr>
<td>Port (port)</td>
<td>The DNS port that Global Server Load Balancer listens to.</td>
</tr>
<tr>
<td>Citrix NetScaler [cmdb_ci_lb_netscaler]</td>
<td></td>
</tr>
<tr>
<td>Serial Number (serial_number)</td>
<td>The serial number of the NetScaler device and its current operational status.</td>
</tr>
<tr>
<td>Operational Status (operational_status)</td>
<td></td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Load Balancer Pool [cmdb_ci_lb_pool]</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>The name of the pool and the method of load balancing it enables.</td>
</tr>
<tr>
<td>Load Balancing Method (load_balancing_method)</td>
<td></td>
</tr>
<tr>
<td>Load Balancer Pool Member [cmdb_ci_lb_pool_member]</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>The name, IP address and service port that each pool member uses.</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Service Port (service_port)</td>
<td></td>
</tr>
<tr>
<td>Load Balancer VLAN [cmdb_ci_lb_vlan]</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>The name of the VLAN and the tag that is associated with this VLAN to identify it.</td>
</tr>
<tr>
<td>Tag (tag)</td>
<td></td>
</tr>
<tr>
<td>Load Balancer Services [cmdb_ci_lb_service]</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>Details about the service running on the load balancer, including</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Port (port)</td>
<td></td>
</tr>
<tr>
<td>Operational Status (operational_status)</td>
<td></td>
</tr>
<tr>
<td>Install Status (install_status)</td>
<td></td>
</tr>
<tr>
<td>Hit count (hit_count)</td>
<td></td>
</tr>
<tr>
<td>Exit Interface Routing Rules [dscy_route_interface]</td>
<td></td>
</tr>
<tr>
<td>Router Interface (router_interface)</td>
<td>Details about the routing rules for the exit interface, including the interface on the router that ...</td>
</tr>
<tr>
<td>Configuration item (cmdb_ci)</td>
<td></td>
</tr>
<tr>
<td>Destination IP network (dest_ip_network)</td>
<td></td>
</tr>
<tr>
<td>Next Hop Routing Rules [dscy_route_next_hop]</td>
<td></td>
</tr>
<tr>
<td>Next hop IP address (router_interface)</td>
<td>Details about the routing rules on the load balancer, including the IP addresses of the next hop router on the network. The Configuration Item field is the router which is the next hop device.</td>
</tr>
<tr>
<td>Router Interface (router_interface)</td>
<td></td>
</tr>
<tr>
<td>Configuration item (cmdb_ci)</td>
<td></td>
</tr>
<tr>
<td>Destination IP network (dest_ip_network)</td>
<td></td>
</tr>
</tbody>
</table>

To see the DNS names and aliases for a NetScaler load balancer, you must configure the form and add the **DNS Names for CIs** related list. DNS names also include aliases.

**Note:** Discovery populates load balancer interface information that is necessary to map the `network path` in Service Mapping.

**Configuration item (CI) Relationships**

These relationships are created to support NetScaler Global Server Load Balancer discovery:
Load balancer: F5 BIG-IP

Discovery and Service Mapping can find F5 BIG-IP load balancers via SNMP, SSH, and through the REST API.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](https://www.servicenow.com/knowledgearticle/KB0694477) for more information.

### Prerequisites and important information

**Important:** Discovery treats hardware load balancers as network devices and attempts to discover them primarily using SNMP. If a load balancer in your system, running on a Linux host, has SNMP and SSH ports open, Discovery might classify it based on the SSH port. This classification has priority over SNMP. To ensure that Discovery properly classifies your hardware load balancers, create a Discovery behavior for load balancers that includes SNMP but not SSH. Software load balancers are treated as applications.

You can download VMware images of BIG-IP with a free 90-day key from [https://www.f5.com/](https://www.f5.com/)

For F5 Global Traffic Manager (GTM) GIB-IP load balancers, Discovery can resolve the DNS name of the F5 GTM hardware as well as the DNS names of all the servers associated with the load balancer that receive distributed traffic. To view this data, navigate to **Configuration > Load Balancers > LB Hardware** and open the F5 load balancer record. Then select the **DNS Names for CIs** related list. **Service Mapping** uses this information to map F5 relationships.

### Credentials

- Configure **SNMP credentials**.

  Note: If you do not want to use SSH credentials, you can use the REST API to create a connection to F5 BIG-IP devices.

- (Optional) If there are iRules or SNMP community credentials are not enough for discovering outgoing connections, configure **SSH credentials** on Now Platform.

  Service Mapping uses the SSH credentials to retrieve connections not from the CMDB. Discovering connections using the SSH protocol is a failover mechanism for the SNMP-based discovery.

- Use **basic authentication** credentials for discovery via REST.

- (Optional) For top-down discovery with Service Mapping, configure **applicative credentials**.

- (Optional) For top-down discovery with Service Mapping, provide a user with permissions to run the following commands:
  - bigpipe commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 9)
  - bigpipe and Traffic Management Shell (TMSH) commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 10)
  - Traffic Management Shell (TMSH) commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 11)
• Traffic Management Shell (TMSH) advanced commands (for BIG-IP LTM F5 or BIG-IP GTM F5 version 10, 11, and 12)
• The `show cm traffic-group - get` command for discovering F5 BIG-IP Device Service Clustering

• (Optional) For top-down discovery with Service Mapping, give the user elevated rights to be able to run Service Mapping commands requiring a privileged user.

F5 cluster discovery

The horizontal discovery process can find F5 clusters. To discover clusters, the credentials must have read permission to traffic groups. For SNMP, this is the regular read permission. For SSH, you need to have Traffic Management Shell (TMSH shell) enabled and accessible.

**Note:** Because F5 clusters share the same virtual IPs and addresses for all devices, the horizontal discovery process cannot find specific IP addresses for each device.

Classifier, patterns, and probes

Discovery uses these classifiers, probes, and pattern for F5 discovery:

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Trigger probes</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>F5 BIG-IP Load Balancer</td>
<td>• Horizontal discovery probe: launches patterns</td>
<td>These patterns are used for standard horizontal discovery including identifying clusters:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F5 Load Balancer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F5 Load Balancer SSH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F5 Load Balancer REST</td>
</tr>
<tr>
<td>F5 HTTP</td>
<td>The F5 HTTP classifier does not use a trigger probe. It triggers the F5 Load Balancer REST pattern to perform data collection via REST.</td>
<td>These patterns use iRules to get information via REST. Service Mapping uses these patterns for top-down discovery:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F5 Load Balancer REST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F5 BigIP GTM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F5 BigIP LTM</td>
</tr>
</tbody>
</table>

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

Connections discovered by Service Mapping during the top-down discovery

Service Mapping performs the top-down discovery of the GTM GIB-IP load balancer in the context of application services. It discovers configured URLs of the GTM GIB-IP load balancers.

Data collected by Discovery during horizontal discovery

The following data is collected on the F5 Big-IP [cmdb_ci_lb_bigip] table.
<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>F5 Big-IP [cmdb_ci_lb_bigip]</strong></td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>Details about the load balancer. View a record in this table to see the</td>
</tr>
<tr>
<td>Serial Number (serial_number)</td>
<td>upstream and downstream relationships with the load balancer.</td>
</tr>
<tr>
<td>Operational Status (operational_status)</td>
<td></td>
</tr>
</tbody>
</table>

| Load Balancer Interface [cmdb_ci_lb_interface] |                                                                             |
| Name (name)                                   | Details about each interface for the load balancer.                        |
| MAC Address (mac_address)                     |                                                                             |
| Operational Status (operational_status)       |                                                                             |

| **Load Balancer Pool [cmdb_ci_lb_pool]**      |                                                                             |
| Name (name)                                   | The name of the pool and the method of load balancing it enables.          |
| Load balancing method (load_balancing_method) |                                                                             |

| **Load Balancer Pool Member [cmdb_ci_lb_pool_member]** |                                                                             |
| Name (name)                                           | The name, IP address, and service port that each pool member uses.          |
| IP Address (ip_address)                              |                                                                             |
| Service Port (service_port)                          |                                                                             |
| Pool (pool)                                          |                                                                             |

| **Load Balancer VLAN [cmdb_ci_lb_vlan]**            |                                                                             |
| Name (name)                                         | The name of the VLAN and the tag that is associated with this VLAN to      |
| Tag (tag)                                           | identify it.                                                               |
| MAC Address (mac_address)                            |                                                                             |

| **Load Balancer Services [cmdb_ci_lb_service]**     |                                                                             |
| Name (name)                                         | Details about the load balancer service that distributes the workload to   |
| IP Address (ip_address)                             | other servers, such as web servers.                                       |
| Port (port)                                         |                                                                             |
| Operational Status (operational_status)             |                                                                             |
| Hit count (hit_count)                               |                                                                             |

| **DNS [cmdb_ci_dns_name]**                          |                                                                             |
| Name (name)                                         | The name and IP address of the Domain Name Server for the network adapters. |
| IP Address (ip_address)                             |                                                                             |

<p>| <strong>Network Adapter [cmdb_ci_network_adapter]</strong>       |                                                                             |
| Name (name)                                         | Details about the network adapters on the load balancer.                   |
| IP Address (ip_address)                             |                                                                             |
| Netmask (netmask)                                   |                                                                             |</p>
<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mac address (mac_address)</td>
<td></td>
</tr>
<tr>
<td>Cluster [cmdb_ci_cluster]</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>Details about the load balancer cluster.</td>
</tr>
<tr>
<td>Cluster status (cluster_status)</td>
<td></td>
</tr>
<tr>
<td>Cluster ID (cluster_id)</td>
<td></td>
</tr>
<tr>
<td>IP address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Cluster Node [cmdb_ci_cluster_node]</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>Details about each node in a cluster.</td>
</tr>
<tr>
<td>Cluster ID (cluster_id)</td>
<td></td>
</tr>
<tr>
<td>Cluster status (cluster_status)</td>
<td></td>
</tr>
</tbody>
</table>

**Configuration item (CI) relationships for clusters**

These relationships are created for clusters:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_cluster</td>
<td>Hosts::Hosted on</td>
<td>cmdb_ci_lb_bigip</td>
</tr>
<tr>
<td>cmdb_ci_cluster_node</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_lb_bigip</td>
</tr>
<tr>
<td>cmdb_ci_cluster_node</td>
<td>Cluster of::Cluster</td>
<td>cmdb_ci_cluster</td>
</tr>
</tbody>
</table>

**Load balancer: HAProxy**

Discovery of HAProxy load balancers is performed by SSH.

*Note:* Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](https:// kb0694477) for more information.

HAProxy is an open-source load balancer that can manage any TCP service. It is particularly suited for HTTP load balancing because it supports session persistence and Layer 7 processing. Discovery supports HAProxy for HTTP load balancing. TCP load-balancing is not supported.

Consider the following requirements for discovering the HAProxy:

- The HAProxy software is installed and running on a Linux server.
- The MID Server is deployed to explore the server and the MID Server has access to the server HAProxy configuration file.
- The configuration probe checks for the `haproxy.cfg` file using one of the following methods:
  - Using the `f` parameter for the HAProxy process output.
  - Using the default `/etc/haproxy/haproxy.cfg` path.
- The HAProxy probes require credentials and execute privileges to run commands.

Discovery uses the **Unix - Active Processes** probe to identify an HAProxy load balancer when the name of the process is `haproxy`. If this criterion matches, a record is created in the HAProxy Load Balancers (cmdb_ci_lb_haproxy) table if one does not already exist for that running process.
The following probes are triggered after classification:

### HAProxy probes

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAProxy – Version</td>
<td>The sensor of this probe populates the HAProxy version in the HAProxy Load Balancers (cmdb_ci_lb_haproxy) table.</td>
<td>haproxy</td>
</tr>
<tr>
<td>HAProxy – Get Configuration</td>
<td>The sensor of this probe populates additional information in the HAProxy Load Balancers (cmdb_ci_lb_haproxy) table. The probe also populates information in the Load Balancer Service (cmdb_ci_lb_service), Load Balancer Pool (cmdb_ci_lb_pool), Load Balancer Pool Member (cmdb_ci_lb_pool_member), Load Balancer Interface (cmdb_ci_lb_interface), and Load Balancer Application (cmdb_ci_lb_apple) tables.</td>
<td>echo, sed, cut, grep, egrep (within the Borne shell script)</td>
</tr>
</tbody>
</table>

In addition to populating the data, the following relationships records are created in CI Relationships (cmdb_rel_ci) table:

- The records in the cmdb_ci_lb_appl table run on the cmdb_ci_web_server table records.
- The records in the cmdb_ci_lb_service table use the cmdb_ci_lb_pool table records.
- The records in the cmdb_ci_pool table are used by the cmdb_ci_service table records.
- The records in the cmdb_ci_pool table are members of the cmdb_ci_pool_member table records.
- The records in the cmdb_ci_pool_member table are members of the cmdb_ci_pool table records.

Load balancer: NGINX

Discovery of NGINX load balancers is performed by SSH.

NGINX is an open source web server with a load balancer. Discovery identifies the web server and information related to the load balancer.

Consider the following requirements for discovering NGINX servers:

- Ensure that the NGINX software is installed and running on the server.
- Grant the MID Server has access to the NGINX configuration file, which is /etc/nginx/nginx.conf by default.
- Enable secure shell (SSH) commands.
- The NGINX probes require credentials and execute privileges to run commands.

For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

The Nginx Process Classifier detects a running process that matches the following criteria during the exploration of a UNIX server:
The name of the process starts with nginx.
The name of the process contains master.

Both of these conditions must match:

- A record is created in the Web Server (cmdb_ci_web_server) table.
- A Runs on relationship is created in the CI Relationship (cmdb_rel_ci) table for the Linux Server (cmdb_ci_linux_server) table and the Web Server (cmdb_ci_web_server) table.

The following probes are triggered after classification:

**NGINX probes**

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nginx – Version</td>
<td>This probe contains a Bourne shell script. It determines the version of NGINX and populates the Web Server (cmdb_ci_web_server) table.</td>
<td>nginx</td>
</tr>
<tr>
<td>Nginx – Get Configuration</td>
<td>This probe contains a Bourne shell script and an argument that determines the path of the NGINX configuration file. The probe identifies configuration parameters based on keywords within the configuration file and returns them as a single payload result.</td>
<td>echo, sed, cut, grep, egrep (within the Bourne shell script)</td>
</tr>
</tbody>
</table>

In addition to populating the data, the following relationships records are created in CI Relationships (cmdb_rel_ci) table:

- The records in the cmdb_ci_web_server table run on the cmdb_ci_linux_server table records.
- The records in the cmdb_ci_lb_service table use the cmdb_ci_lb_pool table records.
- The records in the cmdb_ci_pool table are used by the cmdb_ci_service table records.
- The records in the cmdb_ci_pool table are members of the cmdb_ci_pool_member table records.
- The records in the cmdb_ci_pool_member table are members of the cmdb_ci_pool table records.

Load balancer: Radware-appDirector

Discovery of Radware load balancers is performed by SNMP.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Requirements**

**Important:** Discovery treats hardware load balancers as network devices and attempts to discover them primarily using SNMP. If a load balancer in your system, running on a Linux host, has SNMP and SSH ports open, Discovery might classify it based on the SSH port. This classification has priority over SNMP. To ensure that Discovery properly classifies your...
hardware load balancers, create a Discovery behavior for load balancers that includes SNMP but not SSH. Software load balancers are treated as applications.

Credentials

Configure SNMP credentials.

Classifier, pattern, and probes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probe</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radware - AppDirector - Load Balancer</td>
<td>• Horizontal discovery probe: launches patterns</td>
<td>AppDirector Load Balancer</td>
</tr>
</tbody>
</table>

Data discovery

The following data is collected on the Radware Load Balancers [cmdb_ci_lb_radware] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
</tbody>
</table>

The following data is collected on the DNS Name [cmdb_ci_dns_name] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
</tbody>
</table>

The following data is collected on the Load Balancer Services [cmdb_ci_lb_service] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Port</td>
<td>port</td>
</tr>
<tr>
<td>Operational Status</td>
<td>operational_status</td>
</tr>
<tr>
<td>Install Status</td>
<td>install_status</td>
</tr>
</tbody>
</table>

Load balancer fields and probes

Discovery stores load balancer information in several tables.

The information in this topic pertains to discovery with probes and sensors only. If you are using patterns, such as to discover F5 Big-IP load balancers, see the relevant topic for that type of load balancer.
**Note:** By default, the system uses the discovered IP address of a load balancer for the CI record. This can be the management IP created for the device that is used in the Discovery schedule. For instructions on how to force Discovery to use the IP address of the load balancer’s NIC rather than that of a management IP, see [IP address selection properties](#).

### Fields on the Citrix Netscalers (cmdb_ci_lb_netscaler) table

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number (serial_number)</td>
<td>SNMP - Netscaler - Identity - Serial</td>
</tr>
<tr>
<td>Asset tag (asset_tag)</td>
<td>SNMP - Netscaler - Identity</td>
</tr>
<tr>
<td>Model ID (model_id)</td>
<td>SNMP - Netscaler - Identity</td>
</tr>
</tbody>
</table>

### Fields on the Cisco GSS (cmdb_ci_lb_cisco_gss) table

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number (serial_number)</td>
<td>Cisco GSS - Serial Number</td>
</tr>
<tr>
<td>Model ID (model_id)</td>
<td>Cisco GSS - Identity</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td>Cisco GSS - Network</td>
</tr>
<tr>
<td>Manufacturer (manufacturer)</td>
<td>Cisco GSS - Identity</td>
</tr>
</tbody>
</table>

### Fields on the Cisco CSS (cmdb_ci_lb_cisco_css) table

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number (serial_number)</td>
<td>SNMP - Load Balancer – Identity</td>
</tr>
<tr>
<td>Model ID (model_id)</td>
<td></td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Manufacturer (manufacturer)</td>
<td></td>
</tr>
</tbody>
</table>

### Fields on the F5 BIG-IP (cmdb_ci_lb_bigip) table

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active modules (active_modules)*</td>
<td>SNMP - F5 BIG-IP - System</td>
</tr>
<tr>
<td>Asset tag (asset_tag)</td>
<td></td>
</tr>
<tr>
<td>Model ID (model_id)</td>
<td></td>
</tr>
<tr>
<td>Failover mode (failover_mode)*</td>
<td></td>
</tr>
<tr>
<td>Failover peer (failover_peer)*</td>
<td></td>
</tr>
<tr>
<td>Serial Number (serial_number)</td>
<td>Cisco GSS - Serial Number</td>
</tr>
</tbody>
</table>

*not visible on the form by default. Customize the form to add this field.
**Fields on the Load Balancer (cmdb_ci_lb) table**

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational status (operational_status)</td>
<td>• Apache - Get JK Module</td>
</tr>
<tr>
<td>Name (name)</td>
<td>• Apache - Get Proxy Module</td>
</tr>
<tr>
<td></td>
<td>• HAProxy - Get Configuration</td>
</tr>
<tr>
<td></td>
<td>• SNMP - F5 BIG-IP - System</td>
</tr>
<tr>
<td></td>
<td>• SNMP - Netscaler - System</td>
</tr>
<tr>
<td></td>
<td>• SNMP - Load Balancer - Identity</td>
</tr>
</tbody>
</table>

**Fields on the Load Balancer Application (cmdb_ci_lb_appi) table**

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration file (config_file)*</td>
<td>Nginx - Get Configuration</td>
</tr>
<tr>
<td>Name (name)</td>
<td>• Apache - Get JK Module</td>
</tr>
<tr>
<td>IP Address (ip_address)*</td>
<td>• Apache - Get Proxy Module</td>
</tr>
<tr>
<td>Most recent discovery (last_discovered)*</td>
<td>• HAProxy - Get Configuration</td>
</tr>
<tr>
<td>Version (version)</td>
<td>• Nginx - Get Configuration</td>
</tr>
<tr>
<td></td>
<td>• Apache - Version</td>
</tr>
<tr>
<td></td>
<td>• HAProxy - Version</td>
</tr>
<tr>
<td></td>
<td>• Nginx - Version</td>
</tr>
</tbody>
</table>

**Fields on the Load Balancer Interface (cmdb_ci_lb_interface) table**

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status (install_status)*</td>
<td>• Apache - Get JK Module</td>
</tr>
<tr>
<td>Load Balancer (load_balancer)</td>
<td>• Apache - Get Proxy Module</td>
</tr>
<tr>
<td>Name (name)</td>
<td>• HAProxy - Get Configuration</td>
</tr>
<tr>
<td></td>
<td>• SNMP - F5 BIG-IP - System</td>
</tr>
<tr>
<td></td>
<td>• SNMP - Netscaler - System</td>
</tr>
</tbody>
</table>

*not visible on the form by default. Customize the form to add this field.

**Fields on the Load Balancer Pool (cmdb_ci_lb_pool) table**

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Balancing Method (load_balance_method)</td>
<td>• Apache - Get JK Module</td>
</tr>
<tr>
<td>Load Balancer (load_balancer)</td>
<td>• Apache - Get Proxy Module</td>
</tr>
<tr>
<td>Most recent discovery (last_discovered)*</td>
<td>• HAProxy - Get Configuration</td>
</tr>
<tr>
<td>Name (name)</td>
<td>• SNMP - F5 BIG-IP - System</td>
</tr>
<tr>
<td></td>
<td>• SNMP - Netscaler - System</td>
</tr>
</tbody>
</table>

*not visible on the form by default. Customize the form to add this field.
### Fields for Load Balancer Pool Member (cmdb_ci_lb_pool_member) data

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fully qualified domain name (fqdn)*</td>
<td>• HAProxy - Get Configuration</td>
</tr>
<tr>
<td></td>
<td>• Nginx - Get Configuration</td>
</tr>
<tr>
<td></td>
<td>• SNMP - F5 BIG-IP - System</td>
</tr>
<tr>
<td></td>
<td>• SNMP - Netscaler - System</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Load balancer (load_balancer)</td>
<td></td>
</tr>
<tr>
<td>Most recent discovery (last_discovery)*</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Pool (pool)*</td>
<td></td>
</tr>
<tr>
<td>Service Port (service_port)</td>
<td></td>
</tr>
</tbody>
</table>

*not visible on the form by default. Customize the form to add this field.

### Fields for Load Balancer Service (cmdb_ci_lb_service) data

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input URL (input_url)</td>
<td>• Apache - Get JK Module</td>
</tr>
<tr>
<td></td>
<td>• Apache - Get Proxy Module</td>
</tr>
<tr>
<td></td>
<td>• Nginx - Get Configuration</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Load Balancer (load_balancer)</td>
<td></td>
</tr>
<tr>
<td>Most recent discovery (last_discovery)*</td>
<td></td>
</tr>
<tr>
<td>Service Port (service_port)*</td>
<td></td>
</tr>
</tbody>
</table>

*not visible on the form by default. Customize the form to add this field.
Fields for Load Balancer VLAN (cmdb_ci_lb_vlan) data

<table>
<thead>
<tr>
<th>Field label and name</th>
<th>Probes that gather data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Balancer (load_balancer)</td>
<td>• Apache - Get JK Module&lt;br&gt;• Apache - Get Proxy Module&lt;br&gt;• HAProxy - Get Configuration&lt;br&gt;• Nginx - Get Configuration&lt;br&gt;• SNMP - F5 BIG-IP - System&lt;br&gt;• SNMP - Netscaler - System</td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Tag (tag)</td>
<td>• SNMP - F5 BIG-IP - System&lt;br&gt;• SNMP - Netscaler - System</td>
</tr>
</tbody>
</table>

Apache web server data

Discovery also collects data on Apache web server load balancing modules using SSH. See Apache mod_jk and mod_proxy discovery for more information on probes for Apache web server data. For information on the tables, fields, and data sources that discovery populates for Apache web servers, see Apache web server discovery.

Layer 2 discovery

Discovery can detect the physical connections, known as layer 2, between network devices.

Layer 2 discovery process

Discovery uses multiple probes to gather information about network adapters and their Layer 2 connections. The following diagram shows the probes that Discovery launches for network Discovery, including those used for Layer 2 Discovery.
For example, if Discovery finds a switch in a network, it triggers the **SNMP - Switch - Vlan** probe and the **SNMP - Network - ARPTable** probe. For every Vlan that Discovery finds, it triggers various switch probes. If a switch has routing capabilities, Discovery triggers the **SNMP - Routing** probe to collect network adapter information in the Network Adapter (cmdb_ci_network_adapter) table. If Discovery finds a server, it triggers the appropriate Address Resolution Protocol (ARP) probe for that operating system. Discovery also supports the use of patterns, such as the **Network Switch** and **Network Router** patterns, which are available by default in Discovery. See **Router and switch discovery** for more information.

During the discovery of a network device, Discovery creates records in the Router Interface (dscy_router_interface) table and the Switchport (dscy_switchport) table. This information contains network adapter information for that device. For SNMP-enabled devices, Discovery gathers the information from a routing probe during the exploration phase. The Layer 2 protocol cache probe runs next to collect neighbor data from the device.

### Retrieving neighbor data

As Discovery gathers network information from the probes on a device, the **SNMP - Layer 2 Protocol Caches** identifies device neighbors via CDP and LLDP. This probe updates the Device Neighbors (discovery_device_neighbors) table, which contains the Layer 2 connections between switches. Sometimes, the neighbors of this device might not yet be known to the instance. The neighbor's interface cannot be resolved to a record until Discovery eventually finds the neighbor's side of the relationship. When Discovery runs on the neighboring device, Discovery completes the information for the neighbor's interface for the original reporting device.

Discovery can retrieve neighbor data from these caches on a network device:

- **Cisco Discovery Protocol (CDP):** Cache on Cisco devices that contains device neighbor information in the form of a protocol specific neighbor ID. Cisco Discovery Protocol (CDP).
- **Link Layer Discovery Protocol (LLDP):** Generic cache that contains device neighbor information in the form of a protocol specific neighbor ID.
- **Address Resolution Protocol (ARP):** Cache that contains the IP and MAC addresses of all connecting devices and servers.

### How Layer 2 Discovery works

When device discovery is complete, the system runs the Service Discovery - Device Complete business rule on the Device (discovery_device_history) table. This business rule uses several strategies, together with information from various tables to create **Connects to**: **Connected by** relationships in the CMDB. Relationships can be created between:

- A server and a switch
- Two switches
- A device and the port of another device
- The ports of two devices

In the following example, Discovery found a server running AIX, and was also able to find two IP switches on the network. These relationships were created:

- An **IP Connection** relationship between the AIX Server and the two IP switches A and B.
- A reference between the AIX server and its own network adapter.
- A **Connects to** relationship between the adapters on the two IP switches (not shown in the following image).
- A **Connects to** relationship between the network adapter of the AIX server and the switch port of IP switch A (highlighted in red). This kind of relationship is created by default.

To view these relationships, open the dependency view for the server. To view the relationship between the two IP switches, open the dependency view from one of the switches and select the **Physical Network Connections** option for the **Dependency Type** in the map settings.

**Layer 2 connection strategies**

Layer 2 Discovery uses the following strategies to create relationships between devices in a network:

**PhysicalHostConnectionStrategy**

This strategy creates a connection between a discovered server, which is not a network device, and a network device. This strategy can only create a Layer 2 connection if there is a single MAC address on the port in the switch's forwarding table. If the port has multiple MAC addresses, Discovery does not create a connection between the server and the switch.

If the switch's port is identified, Discovery creates a **Connects to::Connected by** relationship between the server port and the switch port. Otherwise, Discovery creates a **Connects to::Connected by** relationship between the server port and the switch itself.

**VMLayer2ConnectionStrategy**

This strategy creates a connection between a VM and a network device. If the number of exceptional MAC addresses is very low (less than 3 and less than 15% of all MAC address of the ESX server and its VMs), Discovery creates a Layer 2 connection.

**NetworkDeviceLayer2ConnectionStrategy**

This strategy creates a connection between a network device and its neighbors. Discovery creates a Layer 2 connection if there is a single neighbor with a neighbor address, a neighbor interface, or both. If the neighbor's interface exists, Discovery creates a **Connects to::Connected by** relationship between the origin interface and the neighbor's interface. Otherwise, Discovery creates a **Connects to::Connected by** relationship between the origin interface and the switch with the neighbor's address.

**SpanningTreeLayer2ConnectionStrategy**
This strategy creates a connection between a network device and the parent of the network device in the spanning tree.

**JavaScriptLayer2ConnectionStrategy**

This strategy calls a Javascript function with empty implementation. This allows for the creation of an additional strategy.

**Address Resolution Protocol (ARP) in Layer 2 discovery**

The probes for Address Resolution Protocol (ARP) map the IP address of a computer or network device to a MAC address. These probes retrieve the IP address and MAC address for a CI from the Network Infrastructure Item (dscy_net_base) table. Devices that support SNMP, such as Linux computers and network devices, cache two types of address information:

- **Static**: Manually added address resolutions.
- **Dynamic**: Hardware name and IP address pairs added to the cache by previous, successful ARP resolutions.

When the ARP table Discovery completes, the system collects all static and dynamic table entries from devices via SNMP. If a new ARP entry is available, it is added to the Network ARP Table (discovery_net_arp_table). If any previously discovered ARP entries are no longer cached in the device ARP table, the system removes the corresponding records from the CMDB using the reconciliation process.

**Note:** If new ARP entries are created after Discovery runs, they are not discovered until the next Discovery schedule. If ARP entries are removed from the device after Discovery runs, the CMDB ARP table is not updated until Discovery runs again.

Discovery provides these probes for extracting IP and MAC address resolution information:

<table>
<thead>
<tr>
<th>Probe</th>
<th>ECC queue topic</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux - Network ARP Tables</td>
<td>SSHCommand</td>
<td>arp -n</td>
<td>SSH command probe that retrieves the network information from the ARP table on a Linux server.</td>
</tr>
<tr>
<td>Solaris - Network ARP Tables</td>
<td>SSHCommand</td>
<td>arp -an</td>
<td>SSH command probe that retrieves the network information from the ARP table on a Solaris server.</td>
</tr>
<tr>
<td>Windows - Network ARP Table</td>
<td>Powershell</td>
<td>arp -a</td>
<td>Powershell probe that retrieves the network information from the ARP table on a Windows server.</td>
</tr>
</tbody>
</table>
SNMP - Network - ArpTable

The SNMP probe uses this OID first:

iso.org.dod.internet.mgmt.mib-2.ip.ipNetToMediaTable
ipNetToMediaPhysAddress, ipNetToMediaNetAddress.

If the probe fails to return results, it uses this OID:

iso.org.dod.internet.mgmt.mib-2.ip.ipNetToPhysicalTable
ipNetToPhysicalNetAddress, ipNetToPhysicalPhysAddress.

These probes return bridging information from VLANs connected across network switches, including port selection, forwarding tables, and the use of the spanning tree protocol.

SNMP - Switch - BridgePortTable

This probe returns all the ports from a switch that are used to create a bridge between network segments.

Bridging data returned

<table>
<thead>
<tr>
<th>Table</th>
<th>Switch Bridge Port Table (discovery_switch_bridge_port_table) This table is used to map between a port number in the discovery_switch_fwd_table and an interface index.</th>
</tr>
</thead>
<tbody>
<tr>
<td>OID</td>
<td>iso.org.dod.internet.mgmt.mib-2.dot1dBridge.dot1dBase.dot1dBasePortTable, dot1dBasePort, dot1dBasePortIfIndex</td>
</tr>
</tbody>
</table>
| Fields populated | • cmdb_ci  
• port  
• interface_index |

SNMP - Switch - SpanningTreeTable

This probe returns the active path between any two network nodes bridged by a switch.

Spanning tree data returned

<table>
<thead>
<tr>
<th>Table</th>
<th>Switch Spanning Tree Table (discovery_switch_spanning_tree_table). This table is used to find Layer 2 connections between switches.</th>
</tr>
</thead>
<tbody>
<tr>
<td>OID</td>
<td>iso.org.dod.internet.mgmt.mib-2.dot1dBridge.dot1dStp.dot1dStpPortTable, dot1dStpPort, dot1dStpPortState, dot1dStpPortEnable, dot1dStpPortDesignatedRoot, dot1dStpPortDesignatedBridge</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Fields populated
- cmdb_ci
- port
- port_state
- port_enable
- designated_root
- designated_bridge_mac

SNMP - Switch - ForwardingTable

This probe returns information from a switch's forwarding table.

Forwarding table data returned

<table>
<thead>
<tr>
<th>Table</th>
<th>Switch Forwarding Table (discovery_switch_fwd_table) This is the forwarding table for the switches.</th>
</tr>
</thead>
</table>

OIDs
These OIDs are built as needed by the DiscoveryVlanSwitchProcessor script include.

- **Non-Cisco**:  
  - **Q-BRIDGE MIB**: oid_spec_list = 'table iso.org.dod.internet.mgmt.mib-2.dot1dBridge.qBridgeMIB.qBridge qBridgeMIBObjects.dot1qTp.dot1qTpFdbTable dot1qTpFdbAddress,' + vlanIndex + ' ,dot1qTpFdbPort.' + vlanIndex + ' ,dot1qTpFdbStatus.' + vlanIndex;
  - **BRIDGE MIB**: oid_spec_list = 'table iso.org.dod.internet.mgmt.mib-2.dot1dBridge.dot1dTp.dot1dTpFdbTable dot1dTpFdbAddress,dot1dTpFdbPort,dot1dTpFdbStatus';
  - **Cisco BRIDGE MIB**: oid_spec_list = 'table iso.org.dod.internet.mgmt.mib-2.dot1dBridge.dot1dTp.dot1dTpFdbTable dot1dTpFdbAddress,dot1dTpFdbPort,dot1dTpFdbStatus';

Additional probe called
The SSH probe Switch - MAC Table can be triggered by the DiscoveryVlanSwitchProcessor script include to populate additional Layer 2 data. If you are missing Layer 2 data after you run discovery on multiple VLANs, you should enable the triggering of the Switch - MAC Table probe and run discovery again.

To do this, add this property to the System Property (sys_properties) table: glide.discovery.sensors.forwardingtable.ssh. Set the value of the property to true. By default this property is false.

Command
show mac address-table

Fields populated
- cmdb_ci
- vlan_id
- port
- status
- mac_address (from the cmdb_ci field in the Network Infrastructure Item (discovery_net_base) table)

SNMP - Switch - Vlan

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
This probe returns VLAN IDs from a network switch and these OIDs:

- `iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable vtpVlanState`
- `iso.org.dod.internet.private.enterprises.cisco.ciscoMgmt.ciscoVlanMembershipMIB.ciscoVlanMembershipMIBObjects.vmMembership.vmMembershipSummaryTable vmMembershipSummaryVlanIndex,vmMembershipSummaryMemberPorts,vmMembershipSummaryMember2kPorts`
- `iso.org.dod.internet.private.enterprises.juniperMIB.jnxMibs.jnxExMibRoot.jnxExSwitching.jnxExVlan.jnxVlanMIBObjects.jnxExVlanTable jnxExVlanTag`
- `iso.org.dod.internet.mgmt.mib-2.system.sysDescr`

Other switch types are not supported.

**Port tables**

These port tables are populated during horizontal Discovery of a device:

- Network Adapter (`cmdb_ci_network_adapter`)
- Switchport (`dscy_switchport`)
- Router Interface (`dscy_router_interface`)
- Load Balancer Interface (`cmdb_ci_lb_interface`)

**TCP connection discovery**

Discovery identifies and classifies information about TCP connections using the ADM and ADME probes.

Discovery populates this data in the TCP Connections (`cmdb_tcp`) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absent</td>
<td>cmdb_tcp</td>
<td>absent</td>
<td>If the process is no longer running on this device. The first time the process is detected, the value is not selected (meaning that it is not absent). If Discovery runs again and does not detect the process, it is selected.</td>
</tr>
<tr>
<td>Computer</td>
<td>cmdb_tcp</td>
<td>computer</td>
<td>The device that is connected to or is listening on the TCP port.</td>
</tr>
<tr>
<td>IP</td>
<td>cmdb_tcp</td>
<td>ip</td>
<td>The IP address that this device or process is listening on or is connected to. If the IP address is 127.0.0.1, it is the host computer itself.</td>
</tr>
<tr>
<td>PID</td>
<td>cmdb_tcp</td>
<td>pid</td>
<td>The ID of the process that is connecting to or listening on this device (IP address).</td>
</tr>
<tr>
<td>Label</td>
<td>Table Name</td>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Port</td>
<td>cmdb_tcp</td>
<td>port</td>
<td>The port number through which TCP traffic was detected.</td>
</tr>
<tr>
<td>Process</td>
<td>cmdb_tcp</td>
<td>process</td>
<td>The process that is connecting to or listening on the port.</td>
</tr>
</tbody>
</table>
| Type  | cmdb_tcp   | type       | - **Listening on**: the process is listening to the device via TCP.  
|       |            |            | - **Connecting to**: if the process is connected to this device via TCP. |
| count | cmdb_tcp   | count      | The number of times that Discovery detects a specific TCP connection. This field is used by ADM enhanced probes. |

Discovery populates this data in the Running Process (cmdm_running_process) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>The name of the running process.</td>
</tr>
<tr>
<td>Computer</td>
<td>computer</td>
<td>The device that the process is running on.</td>
</tr>
<tr>
<td>Parent</td>
<td>parent</td>
<td>The parent process of this process, if any.</td>
</tr>
<tr>
<td>Listening on</td>
<td>listening_on</td>
<td>The ports that the processes is listening on.</td>
</tr>
<tr>
<td>Connected to</td>
<td>connected_to</td>
<td>The ports that the processes is connected to.</td>
</tr>
<tr>
<td>Command</td>
<td>command</td>
<td>The command for this process, including the path.</td>
</tr>
<tr>
<td>PID</td>
<td>pid</td>
<td>The ID for the process.</td>
</tr>
<tr>
<td>PPID</td>
<td>ppid</td>
<td>The ID for the parent process.</td>
</tr>
<tr>
<td>Absent</td>
<td>absent</td>
<td>If the process is no longer running on this device. The first time the process is detected, the value is not selected (meaning that it is not absent). If Discovery runs again and does not detect the process, it is selected.</td>
</tr>
<tr>
<td>Classify</td>
<td>classify</td>
<td>If the running process should be classified.</td>
</tr>
<tr>
<td>Key parameters</td>
<td>key_parameters</td>
<td>The key parameters for the command.</td>
</tr>
<tr>
<td>Label</td>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Parameters</td>
<td>parameters</td>
<td>The parameters for the command.</td>
</tr>
<tr>
<td>Count</td>
<td>count</td>
<td>The number of times that Discovery detects a specific TCP connection. This field is used by ADM enhanced probes.</td>
</tr>
</tbody>
</table>

### Standard ADM probes

Discovery launches these probes with standard ADM discovery:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Includes these sub-probes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX - ADM</td>
<td>Queries for information about active (running) AIX processes and active connections - the information required to perform application dependency mapping.</td>
<td>AIX - Active Connections, AIX - Active Processes</td>
</tr>
<tr>
<td>HP-UX - ADM</td>
<td>Queries for information about active (running) HP-UX processes and active connections - the information required to perform application dependency mapping. This probe requires that Isol be installed and the UNIX - Active Connections probe be activated, which is inactive by default.</td>
<td>UNIX - Active Connections, HP-UX - Active Processes</td>
</tr>
<tr>
<td>Solaris - ADM</td>
<td>Queries for information about active (running) Solaris processes and active connections - the information required to perform application dependency mapping. This probe is triggered by the Solaris Zones probe. The system triggers an ADM probe for each local zone contained in a global zone. This probe requires that Isol be installed on the global zone.</td>
<td>Solaris - Active Processes, Solaris - Active Connections</td>
</tr>
<tr>
<td>UNIX - ADM</td>
<td>Queries for information about active (running) UNIX processes and active connections - the information required to perform application dependency mapping.</td>
<td>UNIX - Active Connections, UNIX - Active Processes</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Includes these sub-probes</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Windows - ADM               | Queries for information about active (running) Windows processes and active connections - the information required to perform application dependency mapping. | • Windows - Active Connections  
• Windows - Active Processes                                                                 |
| UNIX - ADM netstat          | Performs the same functions as the UNIX ADM probe relying on netstat to support continuous TCP-based discovery. Service Mapping uses this probe for traffic-based discovery. | • UNIX - Active Connections with Netstat. This probe retrieves TCP connections to populate the cmdb_tcp and cmdb_tcp_connection tables.  
• UNIX - Active Processes. This probe retrieves TCP processes and populates the cmdb_running_process table. |

### Enhanced ADM probes

Discovery launches these probes when enhanced ADM is activated:

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows - ADM Enhanced</td>
<td>Performs the same functions as the Windows ADM probe with additional parameters to control continuous TCP-based discovery. This probe is compatible with WinRM and non-WinRM supported MID Servers. Configure your MID Server using the mid.windows.management_protocol parameter. See MID Server parameters for the values you can use for this parameter.</td>
</tr>
<tr>
<td>UNIX - ADM Enhanced</td>
<td>Performs the same functions as the UNIX ADM probe with additional parameters to control continuous TCP-based discovery.</td>
</tr>
</tbody>
</table>

Use these probe parameters with the enhanced ADM probes:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid.adme.max_result_size</td>
<td>Control the size of the TCP tracking file.</td>
</tr>
<tr>
<td>max.adme_result_payload_size</td>
<td>Controls the size of the result, including the payload.</td>
</tr>
</tbody>
</table>

**Note:** If this is greater than the MID server property mid.discovery.max_payload_size, a configuration error can occur.
**IP address discovery**
Discovery identifies and classifies information about IP addresses.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Address</td>
<td>discovery_ip_address</td>
<td>ip_address</td>
<td>Various internal</td>
</tr>
<tr>
<td>Hostname</td>
<td>discovery_ip_address</td>
<td>host_name</td>
<td>Various internal</td>
</tr>
<tr>
<td>Discovered</td>
<td>discovery_ip_address</td>
<td>discovered</td>
<td>Various internal</td>
</tr>
<tr>
<td>Agent</td>
<td>discovery_ip_address</td>
<td>agent</td>
<td>Various internal</td>
</tr>
<tr>
<td>Discoverer</td>
<td>discovery_ip_address</td>
<td>discoverer</td>
<td>Various internal</td>
</tr>
<tr>
<td>In Ranges</td>
<td>discovery_ip_address</td>
<td>in_ranges</td>
<td>Various internal</td>
</tr>
<tr>
<td>MAC Address</td>
<td>discovery_ip_address</td>
<td>mac</td>
<td>Various internal</td>
</tr>
<tr>
<td>CMDB CI</td>
<td>discovery_ip_address</td>
<td>cmdb_ci</td>
<td>Various internal</td>
</tr>
<tr>
<td>SNMP attempted</td>
<td>discovery_ip_address</td>
<td>snmp_attempted</td>
<td>Various internal</td>
</tr>
<tr>
<td>SSH attempted</td>
<td>discovery_ip_address</td>
<td>ssh_attempted</td>
<td>Various internal</td>
</tr>
</tbody>
</table>

**IP network discovery**
Discovery identifies and classifies information about IP networks.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discover</td>
<td>cmdb_ci_ip_network</td>
<td>discover</td>
<td>Various internal</td>
</tr>
<tr>
<td>Subnet</td>
<td>cmdb_ci_ip_network</td>
<td>subnet</td>
<td>Various internal</td>
</tr>
<tr>
<td>Network discovery</td>
<td>cmdb_ci_ip_network</td>
<td>network_discovery</td>
<td>Various internal</td>
</tr>
<tr>
<td>Last discovered</td>
<td>cmdb_ci_ip_network</td>
<td>last_discovered</td>
<td>Various internal</td>
</tr>
<tr>
<td>MID server</td>
<td>cmdb_ci_ip_network</td>
<td>mid_server</td>
<td>Various internal</td>
</tr>
<tr>
<td>Router</td>
<td>cmdb_ci_ip_network</td>
<td>router</td>
<td>Various internal</td>
</tr>
<tr>
<td>State</td>
<td>cmdb_ci_ip_network</td>
<td>state</td>
<td>Various internal</td>
</tr>
</tbody>
</table>

**Network printer discovery**
Discovery identifies and classifies information about network printers.

Discovery can identify and classify Network Printers.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial number</td>
<td>cmdb_ci</td>
<td>serial_number</td>
<td>SNMP, various MIBs</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci</td>
<td>start_date</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci</td>
<td>manufacturer</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci</td>
<td>model_id</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>SNMP, IP MIB</td>
</tr>
</tbody>
</table>
### Router and switch discovery

Discovery identifies and classifies information about network routers and switches.

Network routers and switches often have very similar capabilities. It is very common for some switches known as Layer 3 switches to have IP routing. Larger routers with optional modules might accept switching modules. Because of these overlaps and the resulting ambiguity of a particular device's classification, Discovery collects the same data for both routers and switches, if it is available in any given device. For details on how Discovery collects data about connections between network devices and other components, see [Layer 2 discovery](#).

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Prerequisites for server and network device discovery

- The server or network device needs to have IP Address information.
- The system property `glide.discovery.L3_mapping` is set to true to discover routers and switches.
- The router or Layer 3 switch that provides the IP Address needs to have been successfully discovered with populated Exit Interface Routing Rules.

### System properties

You can configure many aspects of Discovery to work according to your personal preferences. The following properties apply to network routers and switches. To view the properties, navigate to [Discovery Definition > Properties](#) unless indicated otherwise.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>SNMP, IF MIB</td>
</tr>
<tr>
<td>Printer type</td>
<td>cmdb_ci_printer</td>
<td>print_type</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Use count</td>
<td>cmdb_ci_printer</td>
<td>use_count</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Use count units</td>
<td>cmdb_ci_printer</td>
<td>use_units</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Colors</td>
<td>cmdb_ci_printer</td>
<td>colors</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Horizontal resolution</td>
<td>cmdb_ci_printer</td>
<td>horizontal_resolution</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Vertical resolution</td>
<td>cmdb_ci_printer</td>
<td>vertical_resolution</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Resolution units</td>
<td>cmdb_ci_printer</td>
<td>resolution_units</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Description</td>
<td>discovery_printer_supplies</td>
<td>description</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Supply type</td>
<td>discovery_printer_supplies</td>
<td>supply_type</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Supply class</td>
<td>discovery_printer_supplies</td>
<td>supply_class</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Current level</td>
<td>discovery_printer_supplies</td>
<td>current_level</td>
<td>SNMP, PRINT MIB</td>
</tr>
<tr>
<td>Max capacity</td>
<td>discovery_printer_supplies</td>
<td>max_capacity</td>
<td>SNMP, PRINT MIB</td>
</tr>
</tbody>
</table>
### Property Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.discovery.L3_mapping | Provides a logical mapping of the TCP/IP layer for network gears. This is not Layer 2 mapping.  
  - **Type:** string  
  - **Default value:** true  
  - **Location:** System Property (sys_properties) table |
| glide.discovery.max_concurrent_invocations_per_schedule | Sets a maximum number of scheduled invocations of the same Discovery schedule. It prevents a backlog of scheduled runs if Discovery does not finish before the next invocation is scheduled to run. The value is an integer that represents the max number of automated invocations of the same schedule that may proceed at one time. If the limit has been reached, subsequent scheduled invocations are canceled. A value of 0 or any negative number disables this property.  
  - **Type:** integer  
  - **Default value:** 3  
  - **Location:** Discovery Definition > Properties |

### Credentials

Discovery explores many kinds of devices, such as switches, routers, and printers, using the SNMP protocol. SNMP Community Credentials do not include a user name, just a password, which is the community string. The default read-only community string for many SNMP devices is public, and Discovery will try that automatically. SNMPv3 credentials require a user name and a key, depending on the protocol selected. Enter the appropriate SNMP credentials if they differ from the public community string.

**Note:** SSH is not supported for routers and switches, but is supported for load balancers.

### Classifier, pattern, and probes

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
</table>
| Standard Network Router  
Standard Network Switch |  
- Horizontal Pattern: launches patterns  
- SNMP-Routing*  
- DNS*  
- SNMP-Identity*  
- SNMP-Switch-Vlan*  
- SNMP-Switching*  
- SNMP - Network - ArpTable* (Standard Network Switch only) | Network Router  
Network Switch |
*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

^This probe remains active by default for the Standard Network Router classifier, even when Discovery uses pattern discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

**Discovering stacked switches**

The Network Switch shared library supports stacked switches. The CI table for this is IP Switch (`cmdb_ci_ip_switch`). Multiple switches in the stack are identified by the master’s IP address. Each member of the stack is identified by a combination of elements, including the master IP address, a name associated with the switch, and the stack mode. The relationship is created between a master and all of its slaves.

**Router and switch data**

<table>
<thead>
<tr>
<th>Table name</th>
<th>Label/Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci</td>
<td>Serial Number</td>
<td>SNMP, various MIBs</td>
</tr>
<tr>
<td></td>
<td>serial_number</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci</td>
<td>Start Date (only when target is a host)</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td></td>
<td>start_date</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci</td>
<td>Manufacturer</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td></td>
<td>manufacturer</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci</td>
<td>Model ID</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td></td>
<td>model_id</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Name</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Model ID</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td></td>
<td>model ID</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Manufacturer</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td></td>
<td>manufacturer</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Serial Number</td>
<td>SNMP, various MIBs</td>
</tr>
<tr>
<td></td>
<td>serial_number</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>IP Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td></td>
<td>short_description</td>
<td></td>
</tr>
<tr>
<td>Table name</td>
<td>Label/Field name</td>
<td>Source</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Can route IP</td>
<td>SNMP, IP MIB, BGP MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_router</td>
<td>Can switch IP</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Model ID</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Manufacturer</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Serial Number</td>
<td>SNMP, various MIBs</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>IP Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Can route IP</td>
<td>SNMP, IP MIB, BGP MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Ports</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Can switch IP</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Can partition VLANs</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Stack</td>
<td>SNMP, entPhysicalTable</td>
</tr>
<tr>
<td>cmdb_ci_ip_switch</td>
<td>Stack Mode</td>
<td>SNMP, entPhysicalTable</td>
</tr>
<tr>
<td>cmdb_ci_netgear</td>
<td>Can route IP</td>
<td>SNMP, IP MIB, BGP MIB</td>
</tr>
<tr>
<td>cmdb_ci_netgear</td>
<td>Can switch IP</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td>cmdb_ci_netgear</td>
<td>Can partition VLANs</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Table name</th>
<th>Label/Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_netgear</td>
<td>Can Hub</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>can_hub</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_network_adapter</td>
<td>IP Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_network_adapter</td>
<td>MAC Address</td>
<td>SNMP, IF MIB</td>
</tr>
<tr>
<td></td>
<td>mac_address</td>
<td></td>
</tr>
<tr>
<td>discovery_device_neighbors</td>
<td>Neighbor Address</td>
<td>ciscoCdpMIB, lldpMIB</td>
</tr>
<tr>
<td></td>
<td>neighbor_address</td>
<td></td>
</tr>
<tr>
<td>discovery_device_neighbors</td>
<td>Neighbor ID</td>
<td>ciscoCdpMIB, lldpMIB</td>
</tr>
<tr>
<td></td>
<td>neighbor_ID</td>
<td></td>
</tr>
<tr>
<td>discovery_device_neighbors</td>
<td>Neighbor Interface</td>
<td>ciscoCdpMIB, lldpMIB</td>
</tr>
<tr>
<td></td>
<td>neighbor_interface</td>
<td></td>
</tr>
<tr>
<td>discovery_device_neighbors</td>
<td>Neighbor Source</td>
<td>ciscoCdpMIB, lldpMIB</td>
</tr>
<tr>
<td></td>
<td>neighbor_source</td>
<td></td>
</tr>
<tr>
<td>discovery_device_neighbors</td>
<td>Origin Interface</td>
<td>ciscoCdpMIB, lldpMIB</td>
</tr>
<tr>
<td></td>
<td>origin_interface</td>
<td></td>
</tr>
<tr>
<td>discovery_net_arp_table</td>
<td>IP Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>IP_address</td>
<td></td>
</tr>
<tr>
<td>discovery_net_arp_table</td>
<td>MAC Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>mac_address</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_bridge_port_table</td>
<td>Interface Index</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>interface_index</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_bridge_port_table</td>
<td>Port</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>port</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_fwd_table</td>
<td>MAC Address</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>mac_address</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_fwd_table</td>
<td>Port</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>port</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_fwd_table</td>
<td>Status</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_fwd_table</td>
<td>VLAN ID</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>vlan_id</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_spanning_tree_table</td>
<td>Designated Bridge MAC</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>designated_bridge_mac</td>
<td></td>
</tr>
<tr>
<td>Table name</td>
<td>Label/Field name</td>
<td>Source</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>discovery_switch_spanning_tree_table</td>
<td>Designated Root</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>designated_root</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_spanning_tree_table</td>
<td>port</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td>discovery_switch_spanning_tree_table</td>
<td>port Enable</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>port_enable</td>
<td></td>
</tr>
<tr>
<td>discovery_switch_spanning_tree_table</td>
<td>Port State</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>port_state</td>
<td></td>
</tr>
<tr>
<td>dscy_router_interface</td>
<td>Name</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td></td>
</tr>
<tr>
<td>dscy_router_interface</td>
<td>Type</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td></td>
</tr>
<tr>
<td>dscy_router_interface</td>
<td>Number</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td>dscy_router_interface</td>
<td>IP Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>dscy_router_interface</td>
<td>MAC Address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>mac_address</td>
<td></td>
</tr>
<tr>
<td>dscy_route_interface</td>
<td>Destination Network</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>dest_ip_network</td>
<td></td>
</tr>
<tr>
<td>dscy_route_interface</td>
<td>Type</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td></td>
</tr>
<tr>
<td>dscy_route_next_hop</td>
<td>Destination Network</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>dest_ip_network</td>
<td></td>
</tr>
<tr>
<td>dscy_route_next_hop</td>
<td>Type</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td></td>
</tr>
<tr>
<td>dscy_route_next_hop</td>
<td>Next Hop</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td></td>
<td>next_hop_ip_address</td>
<td></td>
</tr>
<tr>
<td>dscy_switchport</td>
<td>Type</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td></td>
</tr>
<tr>
<td>dscy_switchport</td>
<td>Status</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
<tr>
<td>dscy_switchport</td>
<td>MAC Address</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>mac_address</td>
<td></td>
</tr>
<tr>
<td>Table name</td>
<td>Label/Field name</td>
<td>Source</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>dscy_switchport</td>
<td>Port Number</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>port_number</td>
<td></td>
</tr>
<tr>
<td>dscy_switchport</td>
<td>Interface Name</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>interface_name</td>
<td></td>
</tr>
<tr>
<td>dscy_switchport</td>
<td>Interface Number</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>interface_number</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_fwd_rule</td>
<td>MAC Address</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>mac_address</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_fwd_rule</td>
<td>MAC Manufacturer</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>mac_mfr</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_fwd_rule</td>
<td>Status</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_fwd_rule</td>
<td>IP address</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_fwd_rule</td>
<td>Netmask</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>netmask</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Base IP Address</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>base_ip_address</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Base MAC Address</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>base_mac_address</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Base Netmask</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>base_netmask</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Type</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Transparent</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>transparent</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Source Route</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>sourceroute</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Name</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td></td>
</tr>
<tr>
<td>dscy_swch_partition</td>
<td>Status</td>
<td>SNMP, dot1dBridge MIB</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
</tbody>
</table>
Turn off collection of port data for SNMP network devices

In some environments you may not want to collect the port data for SNMP network devices. If the network devices contain an extremely large number of ports and each individual port's information is not critical to the overall management of the devices, you may not want to collect the port data for SNMP network devices. This would affect records from being created in the Switchports table (dscy_switchport). This process only impacts probe discovery.

1. Navigate to Discovery Definition > Probes.
2. Go to the SNMP - Switching probe.
3. In the SNMP Fields tab, find the field named ports. Double-click the value in the Active column. Change the value to false.

The collection and processing of ports information for switches is then disabled. The ports field for the switch CI should have the correct ports count even though the related list switchports is empty.

4. Click the check mark to save the change.

IP service and daemon discovery
Discovery identifies and classifies information about services and daemons.

### Data Collected by Discovery on Services and Daemons

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ip_service</td>
<td>name</td>
<td>Various</td>
</tr>
<tr>
<td>Port</td>
<td>cmdb_ip_service</td>
<td>port</td>
<td>Various</td>
</tr>
<tr>
<td>Description</td>
<td>cmdb_ip_service_ci</td>
<td>description</td>
<td>Various</td>
</tr>
<tr>
<td>CI</td>
<td>cmdb_ip_service_ci</td>
<td>ci</td>
<td>Internal reference</td>
</tr>
<tr>
<td>Service</td>
<td>cmdb_ip_service_ci</td>
<td>service</td>
<td>Internal reference</td>
</tr>
</tbody>
</table>

Uninterruptible power supply (UPS) discovery
Discovery identifies and classifies information about an uninterruptible power supply (UPS).

Discovery can identify and classify uninterruptible power supplies.

### Data collected by Discovery on uninterruptible power supplies

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial number</td>
<td>cmdb_ci</td>
<td>serial_number</td>
<td>SNMP, various MIBs</td>
</tr>
<tr>
<td>Start date</td>
<td>cmdb_ci</td>
<td>start_date</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>cmdb_ci</td>
<td>manufacturer</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>Model ID</td>
<td>cmdb_ci</td>
<td>model_id</td>
<td>SNMP, RFC1213 MIB</td>
</tr>
<tr>
<td>IP address</td>
<td>cmdb_ci_network_adapter</td>
<td>ip_address</td>
<td>SNMP, IP MIB</td>
</tr>
<tr>
<td>MAC address</td>
<td>cmdb_ci_network_adapter</td>
<td>mac_address</td>
<td>SNMP, IF MIB</td>
</tr>
<tr>
<td>UPS software version</td>
<td>cmdb_ci_ups</td>
<td>ups_software_version</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Label</td>
<td>Table name</td>
<td>Field name</td>
<td>Source</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>------------------</td>
<td>--------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>Agent software version</td>
<td>cmdb_ci_ups</td>
<td>agent_software_version</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Attached devices</td>
<td>cmdb_ci_ups</td>
<td>attached_devices</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Battery status</td>
<td>cmdb_ci_ups</td>
<td>battery_status</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Seconds remaining on battery</td>
<td>cmdb_ci_ups</td>
<td>seconds_on_battery</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Estimated minutes remaining on battery</td>
<td>cmdb_ci_ups</td>
<td>est_mins_remaining</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Estimated charge % remaining</td>
<td>cmdb_ci_ups</td>
<td>est_charge_remaining</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Battery voltage</td>
<td>cmdb_ci_ups</td>
<td>battery_voltage</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Battery current</td>
<td>cmdb_ci_ups</td>
<td>battery_current</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Battery temperature (C)</td>
<td>cmdb_ci_ups</td>
<td>battery_temperature</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Input line bads</td>
<td>cmdb_ci_ups</td>
<td>input_line_bads</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Output source</td>
<td>cmdb_ci_ups</td>
<td>output_source</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Output frequency</td>
<td>cmdb_ci_ups</td>
<td>output_freq</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Bypass frequency</td>
<td>cmdb_ci_ups</td>
<td>bypass_freq</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Nominal input voltage</td>
<td>cmdb_ci_ups</td>
<td>nom_input_volt</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Nominal input frequency</td>
<td>cmdb_ci_ups</td>
<td>nom_input_freq</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Nominal output voltage</td>
<td>cmdb_ci_ups</td>
<td>nom_output_volt</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Nominal output frequency</td>
<td>cmdb_ci_ups</td>
<td>nom_output_freq</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Rated output VA</td>
<td>cmdb_ci_ups</td>
<td>rated_output_va</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Rated output power</td>
<td>cmdb_ci_ups</td>
<td>rated_output_power</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Low battery threshold minutes</td>
<td>cmdb_ci_ups</td>
<td>low_battery_threshold_mins</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Audible alarm status</td>
<td>cmdb_ci_ups</td>
<td>audible_alarm_status</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Low voltage transfer point</td>
<td>cmdb_ci_ups</td>
<td>low_voltage_transfer_point</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>High voltage transfer point</td>
<td>cmdb_ci_ups</td>
<td>high_voltage_transfer_point</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Input index</td>
<td>cmdb_ci_ups_input</td>
<td>input_index</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Input frequency (Hz)</td>
<td>cmdb_ci_ups_input</td>
<td>input_freq</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Input voltage (RMS VAC)</td>
<td>cmdb_ci_ups_input</td>
<td>input_volt</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Input current (RMS AAC)</td>
<td>cmdb_ci_ups_input</td>
<td>input_current</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Input power (Watts)</td>
<td>cmdb_ci_ups_input</td>
<td>input_power</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Output index</td>
<td>cmdb_ci_ups_output</td>
<td>output_index</td>
<td>SNMP, UPS MIB</td>
</tr>
<tr>
<td>Output load (%)</td>
<td>cmdb_ci_ups_output</td>
<td>output_load</td>
<td>SNMP, UPS MIB</td>
</tr>
</tbody>
</table>
### IBM WebSphere DataPower discovery

Discovery identifies and classifies information about DataPower devices.

### Pattern

By default, Discovery uses the DataPower Server pattern to perform the discovery.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Credentials

Configure both [SNMP credentials](#) and [applicative credentials](#).

### Classifier and OIDs

The **DataPower Server** classifier uses these OIDs:

<table>
<thead>
<tr>
<th>OID</th>
<th>Manufacturer</th>
<th>Model</th>
<th>Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.6.1.4.1.14685.1.3</td>
<td>IBM</td>
<td>Data Power Hosting Server (cmdb_ci_datapower_server)</td>
<td></td>
</tr>
<tr>
<td>1.3.6.1.4.1.14685.1.8</td>
<td>IBM</td>
<td>Data Power Hosting Server (cmdb_ci_datapower_server)</td>
<td></td>
</tr>
<tr>
<td>1.3.6.1.4.1.14685.1.4</td>
<td>IBM</td>
<td>DataPower XB62</td>
<td>Data Power Hosting Server (cmdb_ci_datapower_server)</td>
</tr>
</tbody>
</table>
The classifier kicks off the *Horizontal Pattern probe* to perform identification and exploration of the device.

### Data collected

The following data is collected on the Data Power Hosting Server \([\text{cmdb\_ci\_datapower\_server}]\) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model name</td>
<td>model_name</td>
</tr>
<tr>
<td>Host name</td>
<td>host_name</td>
</tr>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
</tbody>
</table>

The following data is collected on the Network Adapters \([\text{cmdb\_ci\_network\_adapter}]\) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC address</td>
<td>mac_address</td>
</tr>
<tr>
<td>Name</td>
<td>name</td>
</tr>
</tbody>
</table>

* Cisco Unified Computing System (UCS)-HD device discovery
  Discovery identifies and classifies information about Cisco UCS equipment, including chassis and blades.

  **Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Credentials

Configure both *SNMP credentials* and *applicative credentials*.

### Classifier, pattern, and OIDs

The *UCS* classifier uses *HorizontalDiscoveryProbe* to launch the *UCS - HD* pattern. The classifier also uses these SNMP OIDs, which specify the Cisco UCS Equipment \([\text{cmdb\_ci\_ucs\_equipment}]\) table:

<table>
<thead>
<tr>
<th>OID</th>
<th>Manufacturer</th>
<th>Model</th>
<th>Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.6.1.4.1.9.12.3.1.3.847</td>
<td>Unified Computing Cisco UCS 6120XP 20-Port Fabric Interconnect</td>
<td>1.4(3u)(NX-OS) 5.0 Aggregation Switch</td>
<td>Cisco UCS Equipment</td>
</tr>
<tr>
<td>1.3.6.1.4.1.9.12.3.1.3.899</td>
<td>Unified Computing Cisco UCS 6140XP 40-Port Fabric Interconnect</td>
<td>1.4(3u)(NX-OS) 5.0 Aggregation Switch</td>
<td>Cisco UCS Equipment</td>
</tr>
</tbody>
</table>
Data collected

Only B-series devices are discovered. The UCS - HD pattern does not find C, S, or E series devices. The following data is collected on the Cisco UCS Equipment \texttt{[cmdb\_ci\_ucs\_equipment]} table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Model</td>
<td>model_id</td>
</tr>
<tr>
<td>Distinguished name</td>
<td>dn</td>
</tr>
</tbody>
</table>

The following data is collected on the Cisco UCS Chassis \texttt{[cmdb\_ci\_ucs\_chassis]} table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Model number</td>
<td>model_number</td>
</tr>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
<tr>
<td>Chassis ID</td>
<td>chassis_id</td>
</tr>
<tr>
<td>Operational status</td>
<td>operational_status</td>
</tr>
<tr>
<td>Operability</td>
<td>operability</td>
</tr>
<tr>
<td>Distinguished name</td>
<td>dn</td>
</tr>
</tbody>
</table>

The following data is collected on the Cisco UCS Blade \texttt{[cmdb\_ci\_ucs\_blade]} table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Model number</td>
<td>model_number</td>
</tr>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
<tr>
<td>CPU count</td>
<td>cpu_count</td>
</tr>
<tr>
<td>CPU core count</td>
<td>cpu_core_count</td>
</tr>
<tr>
<td>Ram</td>
<td>ram</td>
</tr>
<tr>
<td>Number of Adapters</td>
<td>number_of_adapters</td>
</tr>
<tr>
<td>Chassis ID</td>
<td>chassis_id</td>
</tr>
</tbody>
</table>

&&

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slot ID</td>
<td>slot_id</td>
</tr>
<tr>
<td>Service ID</td>
<td>server_id</td>
</tr>
<tr>
<td>Description</td>
<td>short_description</td>
</tr>
<tr>
<td>Distinguished name</td>
<td>dn</td>
</tr>
</tbody>
</table>

The following data is collected on the Cisco UCS Rack Mount Units [cmdb_ci_ucs_rack_unit] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor</td>
</tr>
<tr>
<td>Server ID</td>
<td>server_id</td>
</tr>
<tr>
<td>Availability</td>
<td>availability</td>
</tr>
<tr>
<td>UUID</td>
<td>uuid</td>
</tr>
<tr>
<td>Rack serial</td>
<td>rack_serial</td>
</tr>
<tr>
<td>Number of CPUs</td>
<td>num_of_cpus</td>
</tr>
<tr>
<td>Number of cores</td>
<td>num_of_cores</td>
</tr>
<tr>
<td>Number of adapters</td>
<td>num_of_cpus</td>
</tr>
<tr>
<td>Model ID</td>
<td>model_id</td>
</tr>
<tr>
<td>Total Memory</td>
<td>total_memory</td>
</tr>
<tr>
<td>Distinguished name</td>
<td>dn</td>
</tr>
<tr>
<td>State</td>
<td>state</td>
</tr>
</tbody>
</table>

**Relationships**

The following example shows the relationships between a Cisco UCS device, chassis, and blades:
### Cisco UCS

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_ucs_chassis</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_ucs_blade</td>
</tr>
<tr>
<td>cmdb_ci_ucs_equipment</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_ucs_rack_unit</td>
</tr>
<tr>
<td>cmdb_ci_ucs_equipment</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_ucs_chassis</td>
</tr>
</tbody>
</table>
Dell Remote Assistant Card discovery

The Dell™ Remote Assistant Card (DRAC) provides users with tools and functionality to monitor, troubleshoot, and repair servers. You can generate DNS URLs to access out-of-band devices when security prevents access from IP addresses.

To identify the DRAC, Discovery uses the SNMP – DRAC probe. This probe uses SNMPv1 and SNMPv2c. Out-of-band devices have a dedicated management channel for device maintenance. System administrators can remotely monitor and manage devices with the dedicated management channel, including hardware sensor monitoring, access to local media drives, or adjusting BIOS settings. DNS URLs can be generated for out-of-band devices to access the out-of-band device page when the SSL certificate forbids navigating through the IP address.

Data collected

Data collected by Discovery for Dell DRAC

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Name</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firmware version</td>
<td>cmdb_ci_outofband_device</td>
<td>firmware_version</td>
<td>SNMP walk: drsFirmwareVersion</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(racFirmwareVersion for iDRAC7)</td>
</tr>
<tr>
<td>Host *</td>
<td>cmdb_ci_outofband_device</td>
<td>host</td>
<td>SNMP walk: drsSystemServiceTag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(systemServiceTag for iDRAC7)</td>
</tr>
<tr>
<td>IP Address</td>
<td>cmdb_ci_outofband_device</td>
<td>ip_address</td>
<td>DNS probe</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_outofband_device</td>
<td>name</td>
<td>SNMP Identity probe</td>
</tr>
<tr>
<td>Product version</td>
<td>cmdb_ci_outofband_device</td>
<td>product_version</td>
<td>SNMP walk: drsProductVersion</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(racVersion for iDRAC7)</td>
</tr>
<tr>
<td>Type</td>
<td>cmdb_ci_outofband_device</td>
<td>type</td>
<td>SNMP walk: drsProductType</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(racType for iDRAC7)</td>
</tr>
<tr>
<td>URL</td>
<td>cmdb_ci_outofband_device</td>
<td>url</td>
<td>SNMP walk: drsProductURL</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(racURL for iDRAC7)</td>
</tr>
</tbody>
</table>

* Host is a reference to the cmdb_ci_computer table via the serial number. For the cmdb_ci_outofband_device.host field to be populated correctly, the host machine must be discoverable or exist within the CMDB with the appropriate serial number.

Out-of-band devices

A DNS URL is generated for configuration items in the Out-Of-Band Device list if the configuration item has a Name and DNS Domain. The DNS URL enables you to access the out-of-band device page when the SSL certificate forbids navigating through the IP URL. The DNS URL has the format https://Name.DNS_Domain. If either the Name or DNS Domain field is empty, the DNS URL contains the value in the URL field.

HTTP device discovery

Discovery can find devices that use the HTTP protocol.
Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Credentials

Classifier, probes, and patterns

HTTP discovery relies on a the HTTP port probe, which Discovery runs after trying to run SSH and SNMP port probes. This port probe launches the HTTP - Classify probe which runs an HTTP GET request for every entry in the HTTP classification table. You should configure your own classifications that the HTTP - Classify probe can access.

The HTTP port probe uses mappings between a protocol and port number in the IP Service (cmdb_ip_service) table. You can add additional port-to-protocol definitions in this table, or you can override the protocol used for a specific HTTP classifier.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• HTTP (port probe):</td>
<td>HTTP (port probe): this probe launches the HTTP - Classify probe, which runs an HTTP GET request for every entry in the HTTP classification table.</td>
</tr>
</tbody>
</table>

Data collected

Run discovery through an HTTP or HTTPS REST call

Discovery can classify devices using the HTTP(S) protocol. You can create your own HTTP classifier to find devices and access them with Basic Auth credentials, rather than using SNMP or SSH credentials.

Role required: discovery_admin

1. Navigate to Discovery > Credentials.
2. Click New.
3. Select Basic Auth Credentials. In most cases, you can use Basic Auth Credentials to access a device through an HTTP or HTTPS REST call.
4. Add the user name and password you need to access the device.
5. Click Submit.
6. Set up an HTTP classifier by navigating to Discovery Definition > CI Classification > HTTP.
7. Click New.
8. Fill out the form fields (see table):

### The F5 HTTP classification

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the classifier.</td>
</tr>
<tr>
<td>Path</td>
<td>Enter the path for the HTTP GET request.</td>
</tr>
<tr>
<td>Headers</td>
<td>Add an option header with a value. You can use a variable in the format ${variable} instead of a static value. Supported headers depend on the REST web service provider you want to connect to. See the documentation for your web service provider to identify which headers are valid or required.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Port (IP Service)</td>
<td>If you want this classifier to use a custom port that is not already a default value in the IP Service (cmdb_ip_service) table, do the following:</td>
</tr>
<tr>
<td></td>
<td>1. Click the lookup icon to see the list of IP Services.</td>
</tr>
<tr>
<td></td>
<td>2. Click <strong>New</strong> and create a new port-to-protocol mapping.</td>
</tr>
<tr>
<td></td>
<td>3. Select that port-to-protocol mapping for this <strong>Port</strong> field.</td>
</tr>
<tr>
<td></td>
<td>Many commonly used protocols are already mapped to ports by default, such as <strong>http</strong> to port 80 and the <strong>https</strong> to port 443.</td>
</tr>
<tr>
<td></td>
<td>If you want to use one of these default port-to-protocol mappings, do not enter any values in the <strong>Port</strong> field. The HTTP port probe uses the correct one by default.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter a number that represents the priority of this classifier. If Discovery finds more than one classifier that applies, it uses the classifier with lower <strong>Order</strong> number.</td>
</tr>
<tr>
<td>Protocol (optional)</td>
<td>If you configure a unique port-to-protocol mapping for the <strong>Port</strong> field, enter an optional protocol override that Discovery uses with this classifier.</td>
</tr>
<tr>
<td></td>
<td>For example, if you want to use port 5000 and the HTTP protocol, add the 5000/HTTP port-to-protocol mapping to the <strong>Port</strong> field, and then choose <strong>http</strong> in the <strong>Protocol</strong> field.</td>
</tr>
</tbody>
</table>

9. Right-click the header and select **Save**.

   The HTTP Classification Match related list appears. This related list shows criteria that you can configure to match against the body of the response to the GET Request that Discovery sent.

10. In the HTTP Classification Match related list, define the match criteria and specify the pattern to launch for this classifier:
   a) Click **New**.
   b) On the HTTP Classification Match form, fill in the form fields (see table).
c) Click **Submit**.

d) Define as many HTTP Classification Matches as necessary. You might want to do this if there are different patterns for different versions of the device you are trying to discover.

In this example, HTTP classification is configured to use the f5 load balancer API:

If you do not have other types of credentials that can find devices, such as Linux credentials or SNMP credentials, you can see how the Discovery application processes HTTP discovery:

1. Navigate to **Discovery** > **Discovery Schedules** and run a discovery.
2. After the discovery process completes, check the ECC Queue to see the results of the HTTP discovery.

You should see that after the Shazzam probe was launched, and that Unix and SNMP classification probes failed. Both the HTTP - Classify probe and the pattern that you specify in the HTTP classification record should launch.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Name</th>
<th>Source</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiprobe</td>
<td>Unix - Classify</td>
<td>10.196.39.128</td>
<td>input</td>
</tr>
<tr>
<td>Multiprobe</td>
<td>Unix - Classify</td>
<td>10.196.39.128</td>
<td>output</td>
</tr>
<tr>
<td>SNMP</td>
<td>SNMP - Classify</td>
<td>10.196.39.128</td>
<td>output</td>
</tr>
<tr>
<td>SNMP</td>
<td>SNMP - Classify 108 OIDs</td>
<td>10.196.39.128</td>
<td>input</td>
</tr>
<tr>
<td>HTTPClassyProbe</td>
<td>HTTPClassyProbe</td>
<td>10.196.39.128</td>
<td>input</td>
</tr>
<tr>
<td>HTTPClassyProbe</td>
<td>HTTPClassyProbe</td>
<td>10.196.39.128</td>
<td>output</td>
</tr>
<tr>
<td>Shazzam</td>
<td>wmi,snmp,ssh,wins,dns,slp,wbem,vmapp,win..</td>
<td>See Payload</td>
<td>input</td>
</tr>
<tr>
<td>Shazzam</td>
<td>wmi,snmp,ssh,wins,dns,slp,wbem,vmapp,win..</td>
<td>See Payload</td>
<td>output</td>
</tr>
</tbody>
</table>

**Software discovery**

Discovery identifies several types of software.

**Active Directory Domain Controller discovery**

Discovery creates or updates a CMDB record when it detects a running instance of Active Directory Domain Controller on Windows machines.

By default, Discovery uses the **Active Directory Domain Controller On Windows** pattern to perform the discovery.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

The following data is collected on the Active Directory Domain Controllers [cmdb_ci_ad_controller] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
</tbody>
</table>
Adobe JRun discovery
Discovery creates or updates a CMDB record when it detects a running instance of Adobe JRun. By default, Discovery uses the Jrun pattern to perform the Discovery.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Privileged commands
For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

Data collected for Adobe JRun
The following data is collected in the Jrun [cmdb_ci_app_server_jrun] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
</tbody>
</table>

Apache web server discovery
Discovery identifies and classifies information about Apache web servers on both Windows and Linux computers.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Credentials
- Applicative credentials
- Windows credentials
- SSH credentials
For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Classifier, patterns, and probes**

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Server</td>
<td>• Horizontal discovery probe: launches patterns</td>
<td>• Apache On Windows Pattern</td>
</tr>
<tr>
<td></td>
<td>• Apache - Version*</td>
<td>• Apache On Unix Pattern</td>
</tr>
<tr>
<td></td>
<td>• Apache - Get Configuration*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Apache - Get JK Module</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Apache - Get Proxy Module</td>
<td></td>
</tr>
</tbody>
</table>

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

**Data collected**

Discovery collects the following data on the Apache (cmdb_ci_apache_web_server) table for both Unix and Windows:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Description</td>
<td>description</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
</tr>
</tbody>
</table>

Apache mod_jk and mod_proxy discovery

Discovery uses SSH to find Apache connectors mod_jk and mod_proxy.

Discovery uses the Unix - Active Processes probe to identify an Apache web server process by matching one of the following criteria: the name of the process is:

- httpd, or
- apache, or
- httpd2, or
- httpd2-prefork

If there is a match on one of these criteria, a record is created in the Web Server (cmdb_ci_web_server) table if one does not already exist for that running process.

The following probes are triggered after classification:
Apache web server probes

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache – Version</td>
<td>The sensor of this probe populates the Apache version information in the Web Server record.</td>
<td>httpd</td>
</tr>
<tr>
<td>Apache – Get Configuration</td>
<td>This probe contains a Bourne shell script and an argument that determines the path of the Apache configuration file. The sensor of this probe populates some additional information in the Web Server record.</td>
<td>echo, sed, httpd, cut, grep, egrep (within the Bourne shell script)</td>
</tr>
</tbody>
</table>

The sensor processing of the Apache – Get Configuration probe identifies whether either the mod_jk or mod_proxy modules are present and triggers the appropriate probe.

Apache module probes

<table>
<thead>
<tr>
<th>Probe</th>
<th>Description</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache – Get JK Module</td>
<td>If the mod_jk module is running as a load balancer on the server, the sensor of this probe populates the information in the Load Balancer Service (cmdb_ci_lb_service), Load Balancer Pool (cmdb_ci_lb_pool) and Load Balancer Pool Member (cmdb_ci_lb_pool_member) tables.</td>
<td>echo, sed, httpd, cut, grep, egrep (within the Bourne shell script)</td>
</tr>
<tr>
<td>Apache – Get Proxy Module</td>
<td>If the mod_proxy module is running as a load balancer on the server, the sensor of this probe populates the information in the Load Balancer Service (cmdb_ci_lb_service), Load Balancer Pool (cmdb_ci_lb_pool) and Load Balancer Pool Member (cmdb_ci_lb_pool_member) tables.</td>
<td>grep, egrep (within the Bourne shell script)</td>
</tr>
</tbody>
</table>

In addition to data population, the following relationships are created in the CI Relationship (cmdb_rel_ci) table:

- The records in the cmdb_ci_lb_appl table run on the cmdb_ci_web_server table records.
- The records in the cmdb_ci_lb_service table use the cmdb_ci_lb_pool table records.
- The records in the cmdb_ci_pool table are used by the cmdb_ci_service table record.
- The records in the cmdb_ci_pool table are members of the cmdb_ci_pool_member table.
- The records in the cmdb_ci_pool_member table are members of the cmdb_ci_pool table.

Legacy: Add sudo access for the Unix - Active Processes probe

For an Apache web server the Unix - Active Processes probe requires sudo privileges.
Role required: admin

You can configure the Unix - Active Processes probe to elevate privileges.

1. Navigate to Discovery > Probes.
2. In the Search field, search for Unix - Active Processes.
3. Click field, search for Unix - Active Processes probe.
4. In the Probe Parameters related list, click New.
5. Click New
6. Use the following information to fill out the form:

<table>
<thead>
<tr>
<th>Probe</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>must_sudo</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

7. Click Submit

Data collected by Mod_jk module

The Apache mod_jk module forwards requests from the Apache web server to a Servlet container, such as Tomcat.

Additional mod_jk directives can also manage load balancing. Discovery populates the CMDB when it detects an Apache Server. When the Apache Web Server process is detected, if the mod_jk module is running on the web server as a load balancer, the related information populates to the CMDB.

- The MID Server user account to explore the target server must have access to the httpd.conf configuration file in the /etc/httpd/conf/ folder.
- Discovery uses secure shell (SSH) commands to identify the following associated elements:
  - Apache Get Configuration
  - Apache Version
  - Apache Get JK Module
- The following probes require execute privileges to run commands:

<table>
<thead>
<tr>
<th>Probes that require execute privileges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe</td>
</tr>
<tr>
<td>Apache – Get Configuration</td>
</tr>
<tr>
<td>echo, sed, httpd, cut, grep, egrep (within the Borne shell script)</td>
</tr>
<tr>
<td>Apache – Version</td>
</tr>
<tr>
<td>httpd</td>
</tr>
<tr>
<td>Apache – Get JK Module</td>
</tr>
<tr>
<td>echo, sed, httpd, cut, grep, egrep (within the Borne shell script)</td>
</tr>
</tbody>
</table>

Probes and sensors

Discovery uses the Unix - Active Processes probe to identify an Apache server that contains the mod_jk module:

1. The Unix - Active Processes probe detects a running process that matches one of the following criteria:
   - The name of the process is httpd.
• The name of the process is apache.

2. If there is a match on one of these criteria, a record is created in the Web Server table (cmdb_ci_web_server) if one does not already exist for that running process. The following probes are also triggered:

• **Apache – Version**: the sensor of this probe populates the Apache version information in the Web Server record.

• **Apache – Get Configuration**: this probe contains a Bourne shell script and an argument that determines the path of the Apache configuration file. The sensor of this probe populates some additional information in the Web Server record.

3. The sensor processing of Apache – Get configuration probe results triggers the following probes if the mod_jk module is running on the web server:

• **Apache – JK Module**: if the mod_jk module is running as a load balancer on the server, the sensor of this probe populates the information in the Load Balancer Service (cmdb_ci_lb_service), Load Balancer Pool (cmdb_ci_lb_pool) and Load Balancer Pool Member (cmdb_ci_lb_pool_member) tables.

### Data Collected

For the mod_jk module with no load balancer, the following data is collected by default:

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Field and Label Name</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_web_server</td>
<td>Name (name)</td>
<td>Apache – Get Configuration</td>
</tr>
<tr>
<td>cmdb_ci_web_server</td>
<td>Version (version)</td>
<td>Apache – Version</td>
</tr>
<tr>
<td>cmdb_ci_web_server</td>
<td>Description (short_description)</td>
<td>Apache – Version</td>
</tr>
</tbody>
</table>

If the mod_jk module is enabled for load balancing, Discovery connects the following data:

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Field and Label Name</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_lb_appl</td>
<td>Name (name)</td>
<td>Apache – Apache - Get JK Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_appl</td>
<td>IP Address (ip_address)</td>
<td>Apache – Apache - Get JK Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_appl</td>
<td>Last Discovered (last_discovered)</td>
<td>Apache – Apache - Get JK Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>Name (name)</td>
<td>Apache – Apache - Get JK Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>Last Discovered (last_discovered)</td>
<td>Apache – Apache - Get JK Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>IP Address (ip_address)</td>
<td>Apache – Apache - Get JK Module</td>
</tr>
</tbody>
</table>
### Table Name | Field and Label Name | Probe
---|---|---
cmdb_ci_lb_pool_member | Load Balancer (load_balancer) | Apache – Apache - Get JK Module

cmdb_ci_lb_pool_member | Port (port) | Apache – Apache - Get JK Module

cmdb_ci_lb_service | Input URL (input_url) | Apache – Apache - Get JK Module

cmdb_ci_lb_service | Last Discovered (last_discovered) | Apache – Apache - Get JK Module

cmdb_ci_lb_service | IP Address (ip_address) | Apache – Apache - Get JK Module

cmdb_ci_lb_service | Name (name) | Apache – Apache - Get JK Module

cmdb_ci_lb_service | Load Balancer (load_balancer) | Apache – Apache - Get JK Module

cmdb_ci_lb_service | Port (port) | Apache – Apache - Get JK Module

cmdb_ci_lb_pool | Last Discovered (last_discovered) | Apache – Apache - Get JK Module

cmdb_ci_lb_pool | Load balancing Method (load_balancing_method) | Apache – Apache - Get JK Module

cmdb_ci_lb_pool | Load Balancer (load_balancer) | Apache – Apache - Get JK Module

cmdb_ci_lb_pool | Name (name) | Apache – Apache - Get JK Module

### Relationships

In addition to data population, the following relationships are created in the CI Relationship (cmdb_rel_ci) table:

- The records in the cmdb_ci_lb_appl table run on the cmdb_ci_web_server table records.
- The records in the cmdb_ci_lb_service table use the cmdb_ci_lb_pool table records.
- The records in the cmdb_ci_pool table are used by the cmdb_ci_service table record.
- The records in the cmdb_ci_pool table are members of the cmdb_ci_pool_member table.
- The records in the cmdb_ci_pool_member table are members of the cmdb_ci_pool table.

### Data collected by Mod_proxy module

The Apache mod_proxy module implements a proxy, gateway, or cache for the Apache web server.

An additional mod_proxy_balancer can also manage load balancing. Discovery populates the CMDB when it detects an Apache server. When the Apache Web Server process is detected, if the mod_proxy module is running on the web server as a load balancer the related information populates to the CMDB.

Consider the following requirements for discovering an Apache server that contains the mod_proxy module:

- The MID Server user account to explore the target server must have access to the httpd.conf configuration file in the /etc/httpd/conf folder.
• Discovery uses secure shell (SSH) commands to identify the following associated elements:
  • Apache Get Configuration
  • Apache Version
  • Apache Get Proxy Module

• The following probes require execute privileges to run commands:

<table>
<thead>
<tr>
<th>Data collected from the mod_jk module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe</td>
</tr>
<tr>
<td>------------------------</td>
</tr>
<tr>
<td>Apache – Get Configuration</td>
</tr>
<tr>
<td>Apache – Get Proxy Module</td>
</tr>
<tr>
<td>Apache – Version</td>
</tr>
</tbody>
</table>

Probes and sensors

Discovery uses the Unix - Active Processes probe to identify an Apache server that contains the mod_proxy module. The probes and sensors operate in the following manner:

1. The Unix - Active Processes probe detects a running process that matches one of the following criteria:
   • The name of the process is httpd.
   • The name of the process is apache2.

2. If there is a match on one of these criteria, a record is created in the Web Server table (cmdb_ci_web_server) if one does not already exist for that running process. The following probes are also triggered:
   • **Apache – Version**: the sensor of this probe populates the Apache version information in the Web server (cmdb_ci_web_server) record.
   • **Apache – Get Configuration**: this probe contains a Bourne shell script and an argument that determines the path of the Apache configuration file. The sensor of this probe populates some additional information in the Web server (cmdb_ci_web_server) record.

3. The sensor processing of the Apache – Get configuration probe results triggers the following probes if the mod_proxy module is running on the web server:
   • **Apache - Get Proxy Module**: if the mod_proxy module is running as a load balancer on the server, the sensor of this probe populates the information in the Load Balancer Service (cmdb_ci_lb_service), Load Balancer Pool (cmdb_ci_lb_pool) and Load Balancer Pool Member (cmdb_ci_lb_pool_member) tables.

Data Collected

For the mod_proxy module with no load balancer, the following data is collected by default:
Data collected by Discovery by default

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Field and Label Name</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_web_server</td>
<td>Name (name)</td>
<td>Apache – Get Configuration</td>
</tr>
<tr>
<td>cmdb_ci_web_server</td>
<td>Version (version)</td>
<td>Apache – Version</td>
</tr>
<tr>
<td>cmdb_ci_web_server</td>
<td>Description (short_description)</td>
<td>Apache – Version</td>
</tr>
</tbody>
</table>

If the mod_proxy module is enabled for load balancing, Discovery connects the following data:

Data collected by Discovery if the mod_proxy module is identified as a load balancer

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Field and Label Name</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_lb_appl</td>
<td>IP Address (ip_address)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_appl</td>
<td>Last Discovered (last_discovered)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>Name (name)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>Last Discovered (last_discovered)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>IP Address (ip_address)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>Load Balancer (load_balancer)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool_member</td>
<td>Port (port)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Input URL (Input_url)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Last Discovered (last_discovered)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>IP Address (ip_address)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Name (name)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Load Balancer (load_balancer)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Port (port)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Last Discovered (last_discovered)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Load balancing Method (load_balancing_method)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Load Balancer (load_balancer)</td>
<td>Apache - Get Proxy Module</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Name (name)</td>
<td>Apache - Get Proxy Module</td>
</tr>
</tbody>
</table>

Relationships

In addition to data population, the following relationships are created in the CI Relationship (cmdb_rel_ci) table:

- The records in the cmdb_ci_lb_appl table run on the cmdb_ci_web_server table records.
- The records in the cmdb_ci_lb_service table use the cmdb_ci_lb_pool table records.
- The records in the cmdb_ci_pool table are used by the cmdb_ci_service table records.
- The records in the cmdb_ci_pool are members of the cmdb_ci_pool_member table records.
• The records in the cmdb_ci_pool_member is a member of the cmdb_ci_pool table records.

Application Mapping for UNIX discovery
To perform the mapping that establishes application relationships, Discovery must be able to detect TCP connections.

The best way to accomplish this is to run the `lsot` command on target machines. However, this command is not available by default on Solaris and AIX machines. To return TCP connection information on these machines, Discovery uses custom shell scripts. For Solaris, the shell script runs the `pfiles` command to query each process. The `pfiles` command causes momentary pauses in the processes it queries on Solaris target machines. These pauses can last from a few milliseconds to a few seconds, during which no other processing can occur. To avoid pausing process threads, the base system disables active connection probes for Solaris machines, but can be configured to use these if the pauses are not an issue. For AIX machines, Discovery is configured to use the active connection probe for AIX targets. However, the user that executes the shell script on AIX machines might require additional privileges to execute commands, such as `kdb`, employed in the script.

On Linux machines, Discovery uses the `lsot` command (installed by default on Linux) to detect TCP connections. The Linux classification probe triggers the Linux - Active Connections probe, which uses `lsot` to discover application relationships and does not produce any pauses. The `lsot` command is the recommended method of returning active TCP connections and can be installed on Solaris and AIX target machines to eliminate any issues produced by the shell script.

**Note:** Discovery of active connections on HP-UX is not supported.

Using the ServiceNow Shell Script
This procedure causes Discovery to run a shell script that uses the `pfiles` command to query processes on a Solaris machine for active connection information. This procedure does not use the `lsot` command to perform application mapping.

1. Navigate to **Discovery Definition > CI Classification > UNIX**.
2. Select **Solaris** from the list.
3. In the UNIX Classification form for Solaris, select the **Triggers probes** related list, and then click **Edit**.
4. Move **Solaris - Active Connections** from the Collections list to the Triggers probe List.
Using the lsof Command

5. Click Save.
Use the ls of command to return active TCP connections and can be installed on Solaris and AIX target machines.

1. Install lsof on the Solaris and AIX machines on which you want to gather application relationship data.
2. In the instance, navigate to Discovery Definition > CI Classification > UNIX.
3. Select Solaris from the list.
4. In the UNIX Classification form for Solaris, select the Triggers probes related list, and then click Edit.
5. Move Linux - Active Connections from the Collections list to the Triggers probe List. This probe is configured to use lsof.
6. Click Save.
7. Repeat this procedure for the AIX classification probe.
Citrix License Server and Delivery Controller discovery
Discovery can find Citrix License Server instances and Delivery Controller components that run on Windows machines.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Requirements**

- Configure Windows administrative credentials to access the host machine.
- Make sure that PowerShell for Discovery is installed on your MID Servers. The patterns used for Citrix discovery require the ability to run PowerShell commands.
- (Optional) For top-down discovery with Service Mapping, on Citrix Delivery Controller, install the Citrix.Broker module.
- (Optional) For top-down discovery with Service Mapping, a user with permission to run PowerShell commands.
- (Optional) For discovering using the WinRM protocol, see Configure WinRM trusted hosts on MID Server.

**Patterns**

Discovery uses these patterns to perform horizontal discovery:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Process detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citrix License Server</td>
<td>ladmin.exe</td>
</tr>
<tr>
<td>Delivery Controller</td>
<td>BrokerService</td>
</tr>
</tbody>
</table>

**Data collected and relationships**

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citrix License Server</td>
<td>Name (name)</td>
</tr>
<tr>
<td>(cmdb_ci_appl_license_server)</td>
<td>TCP ports (tcp_port)</td>
</tr>
<tr>
<td></td>
<td>Operational status (operational_status)</td>
</tr>
<tr>
<td>Citrix Delivery Controller</td>
<td>Name (name)</td>
</tr>
<tr>
<td>(cmdb_ci_appl_delivery_controller)</td>
<td>Version (version)</td>
</tr>
<tr>
<td></td>
<td>TCP ports (tcp_port)</td>
</tr>
<tr>
<td></td>
<td>Operational status (operational_status)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citrix License Server (cmdb_ci_appl_license_server)</td>
<td>Runs on::Runs</td>
<td>Windows Server (cmdb_ci_win_server)</td>
</tr>
<tr>
<td>Citrix Delivery Controller</td>
<td>Runs on::Runs</td>
<td>Windows Server (cmdb_ci_win_server)</td>
</tr>
</tbody>
</table>
Clustered application discovery on Windows
A process and its corresponding resource information can be used to determine whether the process is a clustered process.

The Microsoft SQL Server process classifier that comes with your instance identifies clustered SQL Servers. You can leverage the same technique to detect other clustered applications on Windows using Discovery.

To detect Windows clustered applications with Discovery, you create a process classifier that detects whether a process is a clustered process. Perform the following steps:

Create a classifier for clustered processes
To detect Windows clustered applications, create a process classifier for that CI type.

1. Navigate to Discovery Definition > CI Classification > Processes.
2. Click New.
3. Fill out the form using the fields from the table.
4. Click Submit.
### Process classification form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the process classifier.</td>
</tr>
<tr>
<td>Table</td>
<td>Select <strong>Application</strong>.</td>
</tr>
<tr>
<td>Relation type</td>
<td>Select the CI relationship type for this classification. The relationship field is only available for Process and Scan Application classifications. Discovery process classifications typically use one of these relationship types:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Runs on::Runs</strong>: Defines the relationship of an application to the host on which it runs. This relationship is expressed from the perspective of the host and the application. For example: My database application runs on server001::server001 runs my database application.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Depends on::Used by</strong>: Defines the relationship of an application that communicates with another application. This relationship is expressed from the perspective of each application. For example: The Tomcat application depends on the MySQL database:: The MySQL database is used by Tomcat.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Virtualized by::Virtualizes</strong>: Defines the relationship of a virtual machine to its host. This relationship is expressed from the perspective of the virtual machine and of the host. For example: server001 is virtualized by ServerESX::Server ESX virtualizes server001.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Cluster of::Cluster</strong>: Defines the relationship of a cluster node to the cluster to which it belongs. This relationship is expressed from the perspective of the cluster node and of the cluster.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Hosted on::Hosts</strong>: Defines the relationship of a cluster node and the Windows host. This relationship is expressed from the perspective of the cluster node and of the Windows host.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this box to activate the process classification. Deselect to deactivate it.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter the order in which Discovery should run this process classification when there are multiple classifications available for a table. Discovery runs process classifications from the lowest to the highest order.</td>
</tr>
<tr>
<td>Test with</td>
<td>Lists the host CI where an automatically generated process classification conditions were met. Use this field to test changes to the process classification to ensure that the updated classification behaves as expected.</td>
</tr>
</tbody>
</table>
## Relate the process classifier to Windows cluster resources

In the second step, relate the process classifier to the information that is in the Windows Cluster resources table.

The **Windows Cluster** resources table is `cmdb_ci_win_cluster_resource`. This relationship is built through JavaScript.

1. Navigate back to the process classification you just made.
2. Click the **Parameters** tab.
3. Click **New**.
4. Fill out the fields on the form as appropriate (see table).
5. Click **Submit**.

### Classifier parameter form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Select a unique name.</td>
</tr>
<tr>
<td>Application</td>
<td>Select <strong>Global</strong>.</td>
</tr>
<tr>
<td>Type</td>
<td>Enter <strong>Cluster</strong>.</td>
</tr>
<tr>
<td>Value</td>
<td>This field defines two JavaScript objects, called <code>resourceType</code> and <code>isMatch</code>. For more information, see Populate the Value field on the Classifier Parameter form.</td>
</tr>
</tbody>
</table>
Populate the Value field on the Classifier Parameter form

The value field defines two JavaScript objects, called `resourceType` and `isMatch`.

The `resourceType` value refers to the Resource Type column in the Windows Cluster Resources table (`cmdb_ci_win_cluster_resource`). The `resourceType` value cannot be empty or null.

1. Navigate to a Windows cluster page to access the Windows Cluster Resources table.
2. Click the Windows Cluster Resources tab.

The `isMatch` value evaluates whether the process is a clustered process.

- An evaluation that returns `true` indicates the process is a clustered application, and the process is classified as a clustered application in the cluster.
- An evaluation that returns `false` classifies the process as a regular application running on the cluster node.

The `isMatch` value is a function that contains two input variables, process and resource:

- `process`: Process is the GlideRecord of the process application. It is determined by the Table field in the classifier. In this example, it is the GlideRecord entry of the Application table (`cmdb_ci_app`) for the process that is being classified. You have access to any field values for the CI type such as name or version.
- `resource`: Resource is the GlideRecord entry in the Windows Cluster Resource table after the resourceType condition has been applied. In the example, it is the GlideRecord entry of the sixth row.

Examples of JavaScript for the Value field

In the following example code: `resourceType: "SQL Server"` the query filters the Windows Cluster Resources table entry for the Resource Type field that has a value that is equal to SQL Server. In the following example table, the sixth record would be returned.

The following script indicates that if there is a resource of type SQL Server, the process is classified as a clustered application.

```javascript
resourceType: "SQL Server"
isMatch: function(process, resource) {
  return true;
}
```

The following script indicates that if there is a resource of type SQL Server, and the application name is equal to the resource name, then the process is classified as a clustered application.

```javascript
resourceType: "SQL Server"
isMatch: function(process, resource) {
  if (process.name == resource.name)
    return true;
}
```

If there are multiple matches to the resourceType condition, the matching function is called multiple times. For the following resourceType example, the matching
function is called twice because there are two entries that have **Physical Disk** in the **Resource Type** column in the sample Windows Cluster Resources table.

```
resourceType: "Physical Disk"
```

**Windows Cluster Resources**

![Windows Cluster Resources table]

---

**Discovery with Software Asset Management**

The table structure for managing software installations behaves differently when the Software Asset Management (SAM) application is activated.

The presence of this application affects the way Discovery stores the software installation data it returns from a scan and require additional configuration within configuration item (CI) records.

The following tables are created when the Software Asset Management plugin is activated:

- Software Discovery Model (cmdb_sam_sw_discovery_model)
- Software Model (cmdb_software_product_model)
- Software Installation (cmdb_sam_sw_install)

**Note:** The original schema used by previous versions is not affected when the Software Asset Management plugin is not activated.

When it runs, Discovery populates the Software Installation (cmdb_sam_sw_install) table. The appropriate configuration item (CI) references the software data. A business rule on this table runs, searching for a matching record on the Software Discovery Model (cmdb_sam_sw_discovery_model) table. If a record exists in the Software Discovery Model table, then the Software Installation table updates the record. If no record exists in the Software Discovery Model table, then one is created.

These tables are **not** used by Discovery with Software Asset Management:
- Software (cmdb_ci_spkg)
- Software Model (cmdb_software_product_model)

**Important:** When SAM is installed, the Software Installation (cmdb_sam_sw_install) table is the appropriate source for all current software data. This means you need to update any related lists or customized reference fields you added to CI records.

Starting with the Kingston release, the Discovery populates the VMWare License Key (samp_vmware_license_key) and VMWare License Key Usage (samp_vmware_license_key_usage) tables. See Data collected for VMware vCenter Server for more information on where Discovery saves vCenter data.

Configure a CI to display Software Asset Management data

After you activate the Software Asset Management plugin, configure the form for any computer CI that you want to discover to show data from the Software Installation (cmdb_sam_sw_install) table.

**Role required:** admin

1. Open any computer CI record.
2. Configure the form to add the Software Installation related list and remove the Software Installed related list.

**Exchange Client Access Server (CAS) discovery**

Discovery creates or updates a CMDB record when it detects a running instance of Exchange Client Access Server.

By default, Discovery uses the CAS On Windows pattern to discover the Exchange Client Access Server.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

**Data collected by Discovery on the Exchange Client Access Server**

The following data is collected on the Exchange Client Access Server (cmdb_ci_exchange_cas) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
</tbody>
</table>

**Exchange Hub Transport Servers discovery**

Discovery creates or updates a CMDB record when it detects a running instance of Exchange Hub.
By default, Discovery uses the **Hub Transport Server On Windows** pattern to discover Exchange Hub Transport Servers.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Data collected by Discovery on Exchange Hub Transport Servers

The following data is collected on the Exchange Hub Transport Servers 
[cmdb_ci_exchange_hub_transport_server] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
</tbody>
</table>

**Exchange MailBox discovery**

Discovery creates or updates a CMDB record when it detects a running instance of Exchange Mailbox.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Prerequisites


The service `winRM` must be running on the host machine. Run the following command in the PowerShell window to verify:

```powershell
```

**Pattern**

By default, Discovery uses the **MailBox On Windows** pattern to discover Exchange MailBox.

**Credentials**

Configure these credentials:
<table>
<thead>
<tr>
<th>Credential type</th>
<th>Additional notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applicative credentials</td>
<td>The user must be able to run Powershell commands against the Exchange hosts.</td>
</tr>
<tr>
<td></td>
<td>Use <strong>Exchange Mailbox</strong> as the CI type for the applicative credential.</td>
</tr>
<tr>
<td></td>
<td><strong>Attention:</strong> Do not use the <strong>Exchange Mailbox server</strong> CI type.</td>
</tr>
<tr>
<td>Windows credentials</td>
<td>The OS user must be a domain user, not a local user for the server.</td>
</tr>
</tbody>
</table>

### Data collected

The following data is collected on the Exchange Mailboxes (cmdb_ci_exchange_mailbox) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
</tbody>
</table>

**General software package discovery**

Discovery identifies and classifies information about general software packages.

**Data collected by Discovery on General Software Packages**

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ci_spkg</td>
<td>name</td>
<td>Various</td>
</tr>
<tr>
<td>Version</td>
<td>cmdb_ci_spkg</td>
<td>version</td>
<td>Various</td>
</tr>
<tr>
<td>Install count</td>
<td>cmdb_ci_spkg</td>
<td>install_count</td>
<td>Various</td>
</tr>
<tr>
<td>License count</td>
<td>cmdb_ci_spkg</td>
<td>license_count</td>
<td>Various</td>
</tr>
<tr>
<td>Microsoft SMS ID</td>
<td>cmdb_ci_spkg</td>
<td>msft_sms_id</td>
<td>Various</td>
</tr>
<tr>
<td>Installed on</td>
<td>cmdb_software_instance</td>
<td>installed_on</td>
<td>Various</td>
</tr>
<tr>
<td>Software</td>
<td>cmdb_software_instance</td>
<td>software</td>
<td>Various</td>
</tr>
</tbody>
</table>

**GlassFish Server discovery**

Discovery creates or updates a CMDB record when it detects a running instance of GlassFish Server.

By default, Discovery uses the following patterns to perform the discovery: **GlassFish Server** and **GlassFish WAR**.
Data collected by Discovery for GlassFish

The following data is collected in the GlassFishe (cmdb_ci_appl_glassfish) and GlassFish Wars (cmdb_ci_appl_glassfish_war) tables:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Port</td>
<td>port</td>
</tr>
</tbody>
</table>

**HP Operations Manager discovery**

Discovery creates or updates a CMDB record when it detects a running instance of HP Operations Manager.

By default, Discovery uses the **HP Operations Manager for Windows** pattern to discover the HP Operations Manager.

*Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.*

Data collected by Discovery for HP Operations Manager

The following data is collected in the HP Operations Manager [cmdb_ci_appl_hp_operations] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
</tbody>
</table>

**HP Service Manager discovery**

Discovery creates or updates a CMDB record when it detects a running instance of HP Service Manager.

By default, Discovery uses the **HP Service Manager Application Server** pattern to discover HP Service Manager.
**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

## Data collected by Discovery for HP Service Manager

The following data is collected on the HP Service Manager `{cmdb_ci_appl_hp_service}` table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor</td>
</tr>
</tbody>
</table>

**IBM WebSphere Message Broker (WMB) and WMB HTTP Listener discovery**

Discovery creates or updates a CMDB record when it detects a running instance of IBM WMB and the WMB HTTP listener for both Linux and Windows.

By default, Discovery uses the following patterns to perform the discovery:

- WMB On Unix
- WMB On Windows
- WMB HTTP Listener On Unix
- WMB HTTP Listener On Windows

**Note:** SNOW OOB WMB patterns support WMB versions 6, 7, and 8.

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

## Data collected by Discovery for IBM WebSphere Message Brokers and IBM WMB Http Listener

The following data is collected in the IBM WebSphere Message Brokers `{cmdb_ci_appl_ibm_wmb}` table and the IBM WMB Http Listener `{cmdb_ci_appl_ibm_wmb_listener}` table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
</tbody>
</table>
IBM Websphere WMQ discovery
Discovery creates or updates a CMDB record when it detects a running instance of IBM Websphere WMQ for Unix and Windows.

By default, Discovery uses the following patterns to perform the discovery:

- WMQ On Unix
- WMQ On Windows

For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Data collected by Discovery for IBM WebSphere WMQ

The following data is collected in the IBM WebSphere WMQ [cmdb_ci_appl_ibm_wmq] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name (Windows only)</td>
<td>fqdn</td>
</tr>
</tbody>
</table>

Microsoft IIS server discovery
Discovery identifies and classifies information about Microsoft IIS servers.

Prerequisites

Install IIS Management Scripts and Tools on a Microsoft IIS Server.

Credentials

Configure these credentials:

- Applicative credentials
- Windows credentials

Classifier, pattern, and probe

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.
Discovery uses these classifiers, probes, and patterns for IIS discovery:

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
</table>
| Microsoft IIS Server | • Horizontal discovery probe: launches patterns  
|                  | • Windows - Get IIS Information*                                               | IIS     |

*Probes inactive on new instances.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

**Data collected by Discovery for IIS Web Server**

The following data is collected in the IIS Web Server (cmdb_ci_microsoft_iis_web_server) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
<td>wmi</td>
</tr>
<tr>
<td>Type*</td>
<td>type</td>
<td></td>
</tr>
<tr>
<td>Running process command</td>
<td>running_process_command</td>
<td></td>
</tr>
<tr>
<td>Running process key parameters</td>
<td>running_process_key_parameters</td>
<td></td>
</tr>
</tbody>
</table>

*The Type is always IIS.

The following data is collected in the Web site (cmdb_ci_web_site) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>wmi</td>
</tr>
<tr>
<td>TCP Port</td>
<td>tcp_port</td>
<td>wmi</td>
</tr>
<tr>
<td>Log directory</td>
<td>log_directory</td>
<td>wmi</td>
</tr>
<tr>
<td>Description</td>
<td>short_description</td>
<td>wmi</td>
</tr>
<tr>
<td>Correlation ID</td>
<td>correlation_id</td>
<td>Internal</td>
</tr>
<tr>
<td>IP address</td>
<td>ip_address</td>
<td>wmi</td>
</tr>
</tbody>
</table>

**Microsoft SharePoint discovery**

Discovery creates or updates a CMDB record when it detects a running instance of Microsoft SharePoint.

By default, Discovery uses the SharePoint pattern to discover Microsoft SharePoint.

*Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.*

**Data collected by Discovery for Microsoft SharePoint**

The following data is collected in the SharePoints [cmdb_ci_appl_sharepoint] table:
<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Category</td>
<td>category</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
</tbody>
</table>

**NGINX web server discovery**

NGINX is an open source web server with a load balancer. Discovery identifies the web server and information related to the load balancer.

Discovery identifies NGINX server software using the following process:

1. The NGINX Process Classifier detects a running process that matches the following criteria during the exploration of a Linux server:
   - Name starts with nginx
   - Name contains master

2. If there is a match:
   - A record is created in the NGINX Web Server (`cmdb_ci_nginx_web_server`) table.
   - A Runs on relationship is created in the CI Relationship (`cmdb_rel_ci`) table for a Linux server (Linux Server (`cmdb_ci_linux_server`)) and for an NGINX web server (NGINX Web Server (`cmdb_ci_nginx_web_server`)).

   The following two probes are triggered:
   - NGINX - Version: This probe contains a Bourne shell script. It determines the version of NGINX and populates the NGINX Web Server (`cmdb_ci_nginx_web_server`) table.
   - NGINX - Get Configuration: This probe contains a Bourne shell script and an argument that determines the path of the NGINX configuration file. The probe identifies configuration parameters based on keywords within the configuration file and returns them as a single payload result.

   The sensor on the ServiceNow instance parses the payload result and populates the CMDB.

You must have the following requirements to discover an NGINX webserver:

- NGINX is installed and running on the server.
- The MID Server has access to the NGINX configuration file, which is `/etc/nginx/nginx.conf` by default.

**Requirements**

For NGINX servers:

- Ensure that the NGINX software is installed and running on the server.
- Grant the MID Server has access to the NGINX configuration file, which is `/etc/nginx/nginx.conf` by default.
- Enable secure shell (SSH) commands to identify the following associated elements:
The following probes require execute privileges to run commands:

<table>
<thead>
<tr>
<th>Probe</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nginx ± Version</td>
<td>nginx</td>
</tr>
<tr>
<td>Nginx ± Get Configuration</td>
<td>echo, sed, httpd, cut, grep, egrep (within the Bourne shell script)</td>
</tr>
</tbody>
</table>

**Probes and Sensors**

Discovery identifies NGINX server software using the following process:

1. The Nginx Process Classifier detects a running process that matches the following criteria during the exploration of a UNIX server:
   - The name of the process starts with `nginx`.
   - The name of the process contains `master`.

2. If there is a match:
   - A record is created in the Web Server (cmdb_ci_web_server) table.
   - A Runs on relationship is created in the CI Relationship (cmdb_rel_ci) table for the Linux Server (cmdb_ci_linux_server) table and the Web Server (cmdb_ci_web_server) table.

3. The following two probes are triggered:
   - **Nginx – Version**: this probe contains a Bourne shell script. It determines the version of NGINX and populates the Web Server (cmdb_ci_web_server) table.
   - **Nginx – Get Configuration**: this probe contains a Bourne shell script and an argument that determines the path of the NGINX configuration file. The probe identifies configuration parameters based on keywords within the configuration file and returns them as a single payload result.

4. The sensor on the ServiceNow instance processes the payload and populates the CMDB.

**Data Collected**

Discovery creates or updates CMDB records when it detects a running NGINX process. The following data is collected.

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Field and Label Name</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ci_lb_appl (name)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>IP Address</td>
<td>cmdb_ci_lb_appl (ip_address)</td>
<td>Nginx– Get Configuration</td>
</tr>
</tbody>
</table>
### Table Name | Field and Label Name | Probe
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Discovered</td>
<td>cmdb_ci_lb_appl (last_discovered)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Version</td>
<td>cmdb_ci_lb_appl (version)</td>
<td>Nginx– Version</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_lb_pool_member (name)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Last Discovered</td>
<td>cmdb_ci_lb_pool_member (last_discovered)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>IP Address</td>
<td>cmdb_ci_lb_pool_member (ip_address)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>cmdb_ci_lb_pool_member (load_balancer)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Fully Qualified Domain Name</td>
<td>cmdb_ci_lb_pool_member (fqdn)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Port</td>
<td>cmdb_ci_lb_pool_member (port)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Last Discovered</td>
<td>cmdb_ci_lb_service (last_discovered)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>IP Address</td>
<td>cmdb_ci_lb_service (ip_address)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_lb_service (name)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>cmdb_ci_lb_service (load_balancer)</td>
<td>Nginx– Get Configuration</td>
</tr>
<tr>
<td>Input URL</td>
<td>cmdb_ci_lb_service (input_url)</td>
<td>NGINX – Get Configuration</td>
</tr>
</tbody>
</table>

### Relationships

In the table (cmdb_rel_ci) the following relationships are populated:
- The records in the cmdb_ci_lb_appl table run on the cmdb_ci_web_server table records.
- The records in the cmdb_ci_lb_service table use the cmdb_ci_lb_pool table records.
- The records in the cmdb_ci_pool table are used by the cmdb_ci_service table records.
- The records in the cmdb_ci_pool table are members of the cmdb_ci_pool_member table.
- The records in the cmdb_ci_pool_member table are members of cmdb_ci_pool table.

**Oracle Tuxedo discovery**

Discovery creates or updates a CMDB record when it detects a running instance of Oracle Tuxedo.

By default, Discovery uses the **Tuxedo** pattern to discover Oracle Tuxedo.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Data collected by Discovery for Oracle Tuxedo

The following data is collected the Tuxedo (cmdb_ci_appl_tuxedo) table:
Puppet automation software discovery
Discovery identifies Puppet Masters running on UNIX servers.
Discovery uses secure shell (SSH) commands to collect information.
To access Puppet Master records, navigate to Configuration > Automation Servers > Puppet Masters. You can view the resources in the related list of the Puppet Master record.

How the Puppet Master probe works
Discovery identifies the Puppet Master using the following process:

1. The UNIX - Active Processes probe detects a running process that matches one of the following criteria:
   - The name of the process is `pe-httpd`.
   - The name of the process is `ruby` and the parameters of the process contain `puppet master`.

2. If there is a match on one of these criteria:
   - A record is created in the Puppet Master table (cmdb_ci_puppet_master) if one does not already exist for that running process.
   - The Puppet - Master Info probe is triggered. The sensor of this probe populates additional information in the Puppet Master (cmdb_ci_puppet_master) record.

3. With the activation of the Puppet Configuration Management plugin, the sensor processing of Puppet - Master Info triggers the following simultaneously:
   - Puppet – Certificate Requests: The sensor of this probe populates the Puppet Certificate Request (puppet_certificate_request table) with open requests. Open requests are requests that are not already signed or rejected.
   - MultiProbe Puppet – Resources: This probe contains the following probes:
     - Puppet – Module: The sensor of this probe populates records within the Puppet Module (puppet_module) table.
     - Puppet – Manifests: The sensor of this probe populates records within the Puppet Manifest (puppet_manifest), Puppet Class (puppet_class), and Puppet Parameter (puppet_parameter) tables.

Requirements to discover a Puppet master
By default, Discovery identifies Puppet Masters running on UNIX servers. Discovery uses secure shell (SSH) commands to collect information.
With the addition of the Puppet Configuration Management plugin, Discovery identifies the following associated elements:

- Puppet Certification Requests
- Puppet Manifests
- Puppet Modules

The credentials used to discover the UNIX server must have privileges to execute the following commands. The use of `sudo` is supported, but you must add the `must_sudo` parameter to the probe.

<table>
<thead>
<tr>
<th>Probe</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Puppet – Master Info</td>
<td>puppet, echo, hostname (within the Borne shell script)</td>
</tr>
<tr>
<td>Puppet – Certificate Requests</td>
<td>puppet</td>
</tr>
<tr>
<td>Puppet – Manifests</td>
<td>echo, sed, find (within the Bourne shell script)</td>
</tr>
<tr>
<td>Puppet – Modules</td>
<td>puppet</td>
</tr>
</tbody>
</table>

Data collected

Data collected by Discovery for Puppet automation software, by default

<table>
<thead>
<tr>
<th>Name</th>
<th>Puppet Master</th>
<th>puppet_module (cmdb_ci_puppet_master)</th>
<th>Puppet - Master Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration directory</td>
<td>Puppet Master</td>
<td>config_directory (cmdb_ci_puppet_master)</td>
<td>Puppet - Master Info</td>
</tr>
<tr>
<td>Manifest directory</td>
<td>Puppet Master</td>
<td>manifestdir (cmdb_ci_puppet_master)</td>
<td>Puppet - Master Info</td>
</tr>
<tr>
<td>Module path</td>
<td>Puppet Master</td>
<td>modulepath (cmdb_ci_puppet_master)</td>
<td>Puppet - Master Info</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>Puppet Master</td>
<td>fqdn (cmdb_ci_puppet_master)</td>
<td>Puppet - Master Info</td>
</tr>
<tr>
<td>IP Address</td>
<td>Puppet Master</td>
<td>ip_address (cmdb_ci_puppet_master)</td>
<td>Puppet - Master Info</td>
</tr>
<tr>
<td>Version</td>
<td>Puppet Master</td>
<td>version (cmdb_ci_puppet_master)</td>
<td>Puppet - Master Info</td>
</tr>
</tbody>
</table>

Data collected by Discovery for Puppet automation software, with the Puppet Configuration Management plugin

<table>
<thead>
<tr>
<th>Name</th>
<th>Module</th>
<th>puppet_module (cmdb_ci_puppet_manifest)</th>
<th>Puppet - Modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path</td>
<td>Module</td>
<td>path</td>
<td>Puppet - Modules</td>
</tr>
<tr>
<td>Name</td>
<td>Manifest</td>
<td>name</td>
<td>Puppet - Manifests</td>
</tr>
<tr>
<td>Path</td>
<td>Manifest</td>
<td>path</td>
<td>Puppet - Manifests</td>
</tr>
</tbody>
</table>
Add the must_sudo parameter to the Puppet probe

To use the Puppet probe with sudo, you must add the must_sudo parameter to each probe that requires it.

Role required: admin

You add the must_sudo parameter for each Puppet probe.

1. Navigate to Discovery > Probes.
2. In the Search field, search on the name Puppet.
3. Click the name of the probe.
   
   For example, Puppet Master - Info.
4. In the Related Links pane, select the Probe parameters tab.
5. Click New.
6. In the Name field, enter must_sudo
7. In the Value field, enter true.
8. Click Submit.

SAP ABAP SAP Central Services (ASCS) discovery

Discovery creates or updates a CMDB record when it detects a running instance of SAP ASCS.

By default, Discovery uses the SAP ASCS Application pattern to discover SAP ASCS.

For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Data collected by Discovery for SAP ASCS

The following data is collected in the SAP ASCS [cmdb_ci_appl_sap_ascs] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Label</td>
<td>Field Name</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Instance name</td>
<td>instance_name</td>
</tr>
<tr>
<td>Instance number</td>
<td>instance_number</td>
</tr>
<tr>
<td>System ID</td>
<td>sid</td>
</tr>
<tr>
<td>System directory</td>
<td>system_directory</td>
</tr>
<tr>
<td>System type</td>
<td>system_type</td>
</tr>
<tr>
<td>Transport domain</td>
<td>transport_domain</td>
</tr>
</tbody>
</table>

### SAP Business Objects CMS Server discovery

Discovery creates or updates a CMDB record when it detects a running instance of SAP Business Objects CMS Server.

By default, Discovery uses the **SAP Business Objects CMS Server on Windows** pattern to discover SAP Business Objects CMS Server.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Data collected by Discovery for SAP Business Objects CMS Server

The following data is collected on the SAP Business Objects CMS server [cmdb_ci_appl_sap_bo] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Server Name</td>
<td>server_name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
</tbody>
</table>

### SAP Central Services discovery

Discovery creates or updates a CMDB record when it detects a running instance of SAP Central Services (SAS).

By default, Discovery uses the **SAP SCS Application** pattern to discover SAP Central Services.

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.
### Data collected by Discovery for SAP Central Services

The following data is collected in the SAP SCS Application [cmdb_ci_appl_sap_scs] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Instance name</td>
<td>instance_name</td>
</tr>
<tr>
<td>Instance number</td>
<td>instance_number</td>
</tr>
<tr>
<td>System ID</td>
<td>sid</td>
</tr>
<tr>
<td>System directory</td>
<td>system_directory</td>
</tr>
<tr>
<td>System type</td>
<td>system_type</td>
</tr>
<tr>
<td>Transport domain</td>
<td>transport_domain</td>
</tr>
</tbody>
</table>

### SAP Convergent Invoicing (CI) discovery

Discovery creates or updates a CMDB record when it detects an instance of SAP CI.

By default, Discovery uses the **SAP CI Application** pattern to discover SAP Convergent Invoicing.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](https:// servicenow.com) for more information.

### Data collected by Discovery for SAP Convergent Invoicing

The following data is collected in the SAP CI Application [cmdb_ci_appl_sap_ci] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Instance name</td>
<td>instance_name</td>
</tr>
<tr>
<td>Instance number</td>
<td>instance_number</td>
</tr>
<tr>
<td>System ID</td>
<td>sid</td>
</tr>
<tr>
<td>System directory</td>
<td>system_directory</td>
</tr>
<tr>
<td>System type</td>
<td>system_type</td>
</tr>
<tr>
<td>Transport domain</td>
<td>transport_domain</td>
</tr>
</tbody>
</table>
SAP Development Infrastructure (DI) discovery
Discovery creates or updates a CMDB record when it detects a running instance of SAP DI.

By default, Discovery uses the SAP DI Application pattern to discover SAP Development Infrastructure.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Data collected by Discovery for SAP Development Infrastructure

The following data is collected in the SAP DI Applications [cmdb_ci_appl_sap_di] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Instance name</td>
<td>instance_name</td>
</tr>
<tr>
<td>Instance number</td>
<td>instance_number</td>
</tr>
<tr>
<td>System ID</td>
<td>sid</td>
</tr>
<tr>
<td>System directory</td>
<td>system_directory</td>
</tr>
<tr>
<td>System type</td>
<td>system_type</td>
</tr>
<tr>
<td>Transport domain</td>
<td>transport_domain</td>
</tr>
</tbody>
</table>

SAP Evaluated Receipt Settlement (ERS) discovery
Discovery creates or updates a CMDB record when it detects a running instance of SAP ERS.

By default, Discovery uses the SAP ERS Application pattern to discover SAP Evaluated Receipt Settlement.

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Data collected by Discovery for SAP Evaluated Receipt Settlement

The following data is collected in the SAP ERS Application [cmdb_ci_appl_sap_ers] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
</tbody>
</table>
SQL Server Analysis Services (SSAS) discovery
Discovery creates or updates a CMDB record when it detects a running instance of SQL Server Analysis Services.

By default, Discovery uses the **SSAS pattern** pattern to discover SQL Server Analysis Services.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

Data collected by Discovery for SQL Server Analysis Services

The following data is collected in the SQL Server Analysis Services 
[cmdb_ci_db_mssql_analysis] table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance name</td>
<td>instance_name</td>
</tr>
<tr>
<td>Instance number</td>
<td>instance_number</td>
</tr>
<tr>
<td>System ID</td>
<td>sid</td>
</tr>
<tr>
<td>System directory</td>
<td>system_directory</td>
</tr>
<tr>
<td>System type</td>
<td>system_type</td>
</tr>
<tr>
<td>Transport domain</td>
<td>transport_domain</td>
</tr>
</tbody>
</table>

Sun Java Enterprise System (JES) discovery
Discovery creates or updates a CMDB record when it detects a running instance of Sun JES.

By default, Discovery uses the **Sun JES pattern** pattern to discover Sun Java Enterprise System.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

Data collected by Discovery for Sun Java Enterprise System

The following data is collected in the JES [cmdb_ci_email_server_jes] table:
**Tibco ActiveMatrix BusinessWorks discovery**
Discovery creates or updates a CMDB record when it detects a running instance of Tibco ActiveMatrix BusinessWorks.

By default, Discovery uses the following patterns to perform the discovery:

- **ActiveMatrix Business Works**
- **ActiveMatrix Business Works Process**

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Data collected by Discovery for Tibco ActiveMatrix BusinessWorks**

The following data is collected in the ActiveMatrix Business Works `[cmdb_ci_appl_tibco_matrix]` table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>Project</td>
<td>project</td>
</tr>
</tbody>
</table>

**Tibco Enterprise Message Service discovery**
Discovery creates or updates a CMDB record when it detects a running instance of Tibco Enterprise Message Service.

By default, Discovery uses the **Enterprise Message Service** pattern to discover Tibco Enterprise Message Service.

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.
Data collected by Discovery for Tibco Enterprise Message Service

The following data is collected in the Tibco Enterprise Message Services [cmdb_ci_appl_tibco_message] table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
</tr>
<tr>
<td>Project</td>
<td>project</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>config_directory</td>
</tr>
<tr>
<td>Configuration file</td>
<td>config_file</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
</tbody>
</table>

**Tomcat server discovery**

Discovery identifies and classifies information about Tomcat servers and Web applications running on Unix machines.

Discovery can identify and classify Web applications present in either the CATALINA_BASE directory or the CATALINA_HOME directory.

The classifier that finds Tomcat server processes uses the condition: (Parameters) (contains) (org.apache.catalina.startup.Bootstrap).

**Credentials**

Configure these credentials:

- **Applicative credentials**
- **SSH credentials**

For a list of privileged commands that you need for Discovery and Service Mapping, see *Service Mapping commands requiring a privileged user* for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.
### Classifier, pattern, and probes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tomcat</td>
<td>• Horizontal pattern: launches patterns&lt;br&gt;• SSHCommand-Tomcat - Find web.xml List*&lt;br&gt;• SSHCommand-Tomcat - Get server.xml*&lt;br&gt;• SSHCommand-Tomcat - Get Version*</td>
<td>Tomcat</td>
</tr>
</tbody>
</table>

*Probes inactive on new instances.*

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

### Data collected

Discovery collects the following data on the Tomcat (cmdb_ci_app_server_tomcat) table for both Unix and Windows:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
<td>server.info</td>
</tr>
<tr>
<td>Install Directory</td>
<td>install_directory</td>
<td>server.xml</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
<td>server.info</td>
</tr>
<tr>
<td>Server Port</td>
<td>server_port</td>
<td>server.xml</td>
</tr>
<tr>
<td>Attributes</td>
<td>attributes</td>
<td>server.xml</td>
</tr>
</tbody>
</table>

The following data is collected on the Tomcat Connector (cmdb_ci_tomcat_connector) table for both Unix and Windows:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tomcat</td>
<td>tomcat</td>
<td>server.xml</td>
</tr>
<tr>
<td>Port</td>
<td>port</td>
<td>server.xml</td>
</tr>
</tbody>
</table>

The following data is collected on the Web Service (cmdb_ci_web_service) table for both Unix and Windows:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>App Server</td>
<td>app_server</td>
<td>Initial reference</td>
</tr>
</tbody>
</table>

The following data is collected on the Web Application (cmdb_ci_web_application) table for both Unix and Windows:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>description</td>
<td>web.xml</td>
</tr>
</tbody>
</table>
WebLogic application server discovery

Discovery creates or updates a CMDB record when it detects an instance of an Oracle or BEA Weblogic application server running on a Windows or Linux system.

Requirements

For Windows

- **PowerShell** must be enabled on the MID Server.
- The WebLogic Administration Server instances that start via WebLogic NodeManager must have the `-Dweblogic.RootDirectory=<path>` parameter defined upon server startup. The Windows credential must also have read permissions on the `config.xml` file.

For Linux

- Discovery requires the use of these Bourne shell commands:
  - `find`
  - `cat`
  - `dirname`

- WebLogic administration server instances started via NodeManager must have the `-Dweblogic.RootDirectory=<path>` parameter defined and visible through the Linux `ps` `process` `stat` command (for each AdminServer) for the rest of the Linux WebLogic application server and web application information to be populated in the CMDB.
- The following configuration must be set on the `sudoers` file:
  ```
  User ALL=(root) NOPASSWD: /bin/sh *
  User is the user name on the SSH credential that you create for this discovery.
  ```
- This command must be able to be run:
  ```
  /bin/sh /tmp/snc-*--findcat.sh * *
  ```
- Do not make any changes to `findcat.sh`.
- If you are using probes, not patterns, for Weblogic discovery, the **Linux - Weblogic - Find config.xml** probe must specify this parameter and value: **must_sudo = true**

Note: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.
Credentials

- Applicative credentials
- Windows credentials
- SSH credentials

Classifier, pattern, and probes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weblogic Server</td>
<td>• Horizontal Pattern: launches patterns&lt;br&gt; • Windows - Weblogic - Get web.xml*&lt;br&gt; • Windows - Weblogic - Find config.xml*&lt;br&gt; • Linux - Weblogic - Find config.xml*</td>
<td>WebLogic</td>
</tr>
</tbody>
</table>

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Data collected on Windows

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_cl_app_server_weblogic</td>
<td>name</td>
<td>running process</td>
</tr>
<tr>
<td>Version</td>
<td>cmdb_cl_app_server_weblogic</td>
<td>version</td>
<td>config.xml</td>
</tr>
<tr>
<td>Weblogic domain</td>
<td>cmdb_cl_app_server_weblogic</td>
<td>weblogic_domain</td>
<td>config.xml</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>cmdb_cl_app_server_weblogic</td>
<td>tcp_port</td>
<td>web.xml and running process</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_cl_web_application</td>
<td>name</td>
<td>config.xml</td>
</tr>
<tr>
<td>Document base</td>
<td>cmdb_cl_web_application</td>
<td>document_base</td>
<td>config.xml</td>
</tr>
<tr>
<td>Description</td>
<td>cmdb_cl_web_application</td>
<td>description</td>
<td>web.xml</td>
</tr>
<tr>
<td>Servlet class</td>
<td>cmdb_cl_web_application</td>
<td>servlet_class</td>
<td>web.xml</td>
</tr>
<tr>
<td>Servlet name</td>
<td>cmdb_cl_web_application</td>
<td>servlet_name</td>
<td>web.xml</td>
</tr>
<tr>
<td>App server</td>
<td>cmdb_cl_web_application</td>
<td>app_server</td>
<td>config.xml</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>cmdb_cl_web_application</td>
<td>tcp_port</td>
<td>web.xml</td>
</tr>
</tbody>
</table>
Data collected on Linux

<table>
<thead>
<tr>
<th>Table</th>
<th>Field</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>Name (name)</td>
<td>running process</td>
</tr>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>TCP port (tcp_port)</td>
<td>running process</td>
</tr>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>Version (version)</td>
<td>config.xml</td>
</tr>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>Weblogic domain (weblogic_domain)</td>
<td>config.xml</td>
</tr>
<tr>
<td>cmdb_ci_web_application</td>
<td>Name (name)</td>
<td>config.xml</td>
</tr>
<tr>
<td>cmdb_ci_web_application</td>
<td>Context path (context_path)</td>
<td>config.xml</td>
</tr>
<tr>
<td>cmdb_ci_web_application</td>
<td>App server (app_server)</td>
<td>config.xml</td>
</tr>
</tbody>
</table>

Relationships

<table>
<thead>
<tr>
<th>Parent class</th>
<th>Relationship type</th>
<th>Child class</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_windows_server or cmdb_ci_linux_server</td>
</tr>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_web_application</td>
</tr>
<tr>
<td>cmdb_ci_app_server_weblogic</td>
<td>Depends on::Used by</td>
<td>cmdb_ci_app_server_weblogic</td>
</tr>
</tbody>
</table>

This relationship is made between an AdminServer and any managed servers it encapsulates.
Add sudo access for the Weblogic - Find config.xml probe

If you are not using the Weblogic pattern, you can add sudo privileges to the Linux, the Weblogic - Find config.xml probe.

Role required: admin

1. Navigate to **Discovery Definition > Probes**.
2. Search for and select the **Linux - Weblogic - Find config.xml** probe.
3. In the **Probe Parameters** related list, click **New**
4. Use the following information to complete the form

<table>
<thead>
<tr>
<th>Probe</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>must_sudo</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>
5. Click Submit

Identifying Windows WebLogic application servers

If you are not using the Weblogic pattern, Discovery follows this process when it uses the Windows - Active Processes or Linux - Active Processes probes.

This process describes the use of probes, which are active on your instance if you upgrade. For new instances, Discovery uses the Weblogic pattern.

1. The Windows - Active Processes probe or Linux - Active Processes detects a running process that matches one of the following criteria:
   - The parameters of the process contain `weblogic.Server`.
   - The parameters of the process contain `-Dweblogic.name`.

2. If there is a match on one of the criteria:
   - A record is created in the BEA Weblogic (cmdb_ci_app_server_weblogic) table. The record is populated with the server name and TCP port, which is gathered from the running process.
   - The Windows - Weblogic - Find config.xml probe or the Linux - Weblogic - Find config.xml probe triggers. The sensor of this probe populates additional information in the BEA Weblogic (cmdb_ci_app_server_weblogic) record and the Web Application (cmdb_ci_web_application) record if applicable.

3. The Windows - Weblogic - Find config.xml probe or the Linux - Weblogic - Find config.xml probe attempts to find the related `config.xml` file for the server by:
   - Using the `-Dweblogic.RootDirectory=<path>` parameter defined in the running process.
   - (Windows) Searching for `config.xml` files under the `-Dplatform.home=<path>` parameter defined in the running process (not as efficient using the parameters of the process).
   - (Linux) Searching for the parent process that started the WebLogic server (only viable if the weblogic jvm was started via the `startWeblogic.sh` or related custom script and not the init process).

4. If there are associated web applications found in the WebLogic `config.xml` file, the Windows – Weblogic find web.xml probe triggers for each application. This probe reads the WebLogic `web.xml` file for each web application and the sensor, and then populates additional information.

**WebSphere server discovery**

The IBM WebSphere application server is a software framework with middleware that hosts Java-based web applications. Discovery creates or updates a CMDB record when it detects an instance of a WebSphere application server running on a Windows or Linux system.

**WebSphere discovery on Windows**

Requirements for discovery:

For WebSphere application servers running on Windows systems, enable PowerShell on the MID Server.
Linux discovery on Windows

Requirements for discovery:

- Enable SSH on the WebSphere Application Server.
- Set execute privileges to enable the following probes to run commands:
  - WebSphere - Get serverindex.xml: cat/read permissions on the serverindex.xml file
  - WebSphere - Get cell.xml: cat/read permissions on the cell.xml file
  - WebSphere - Get server.xml: cat/read permissions on the server.xml file

For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

Probes and sensors

Discovery uses this process to identify Linux WebSphere application servers:

1. The Unix - Active Processes probe detects a running process that matches the com.ibm.ws.runtime.WsServer. For example, the ...
   com.ibm.ws.bootstr...com.ibm.ws.runtime.WsServer process output has a parameter of /opt/IBM/WPS/profiles/ccmdb/config localhostNode01Cell ccmdb01 server1. The values are:
   - last parameter = server_name = server1
   - last parameter - 1 = node_name = ccmdb01
   - last parameter - 2 = cell_name = localhostNode01Cell
   - last parameter - 3 = config_path = '/opt/IBM/WPS/profiles/ccmdb/config'

2. If there is a match, a record is created in the IBM Websphere (cmdb_ci_app_server_websphere) table. The following probes are triggered:
   - WebSphere - Cell
   - WebSphere - Web Applications
   - WebSphere - Web Services

3. The WebSphere - Cell probe searches for the cell.xml file for the instance by using the parameters in the running process, and then searching in the related \<config_path>\cells \<cell_name>\ directory.

4. If the probe successfully finds the cell.xml file, the sensor reads its contents and populates additional Websphere Cell (cmdb_ci_websphere_cell) table records as necessary.

5. If the probe successfully finds the serverindex.xml file, the sensor reads its contents and populates additional Web Application (cmdb_ci_web_application) table records as necessary.

6. If the probe successfully finds the server.xml file, the sensor reads its contents and populates additional Web Service (cmdb_ci_web_service) table records as necessary.

Discovery uses this process to identify Windows WebSphere application servers:

1. The Windows - Active Processes probe detects a running process that matches the com.ibm.ws.runtime.WsServer. For example, the ...
   com.ibm.ws.bootstr...com.ibm.ws.runtime.WsServer process output has a parameter of /opt/IBM/WPS/profiles/ccmdb/config localhostNode01Cell ccmdb01 server1. The values are:
2. If there is a match, a record is created in the IBM WebSphere (cmdb_ci_app_server_websphere) table. The following probes are triggered:
   - Windows - WebSphere - Cell
   - Windows - WebSphere - Web Applications
   - Windows - WebSphere - Web Services

3. The Windows - WebSphere - Cell probe searches for the cell.xml file for the instance by using the parameters in the running process, and then searching in the related <config_path>/cells/<cell_name>/ directory.

4. If the probe successfully finds the cell.xml file, the sensor reads its contents and populates additional WebSphere Cell (cmdb_ci_websphere_cell) table records as necessary.

5. The Windows - WebSphere - Web Applications probe searches the serverindex.xml file for the instance by using the parameters in the running process, and then searching in the related <config_path>/cells/<cell_name>/nodes/<node_name>/ directory.

6. If the probe successfully finds the serverindex.xml file, the sensor reads its contents and populates additional Web Application (cmdb_ci_web_application) table records as necessary.

7. The Windows WebSphere - Web Services probe searches for the server.xml file for the instance by using the parameters in the running process, and then searching in the related <config_path>/cells/<cell_name>/nodes/<node_name>/servers/<server_name>/ directory.

8. If the probe successfully finds the server.xml file, the sensor reads its contents and populates additional Web Service (cmdb_ci_web_service) table records as necessary.

Data collected

Data collected on Windows Websphere servers

<table>
<thead>
<tr>
<th>Table</th>
<th>Field</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_app_server_websphere</td>
<td>Name (name)</td>
<td>Running process</td>
</tr>
<tr>
<td>cmdb_ci_websphere_cell</td>
<td>Name (name)</td>
<td>cell.xml</td>
</tr>
<tr>
<td>cmdb_ci_websphere_cell</td>
<td>Cell ID (cell_id)</td>
<td>cell.xml</td>
</tr>
<tr>
<td>cmdb_ci_websphere_cell</td>
<td>Cell type (cell_type)</td>
<td>cell.xml</td>
</tr>
<tr>
<td>cmdb_ci_websphere_cell</td>
<td>Cell discovery protocol</td>
<td>cell.xml</td>
</tr>
<tr>
<td>cmdb_ci_web_service</td>
<td>Name (name)</td>
<td>server.xml</td>
</tr>
<tr>
<td>cmdb_ci_web_service</td>
<td>Service ID (service_id)</td>
<td>server.xml</td>
</tr>
<tr>
<td>cmdb_ci_web_application</td>
<td>Name (name)</td>
<td>serverindex.xml</td>
</tr>
</tbody>
</table>
Data collected on Linux Websphere servers

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ci_websphere_cell</td>
<td>name</td>
<td>server.xml</td>
</tr>
<tr>
<td>App server</td>
<td>cmdb_ci_web_service</td>
<td>app_server</td>
<td>Internal reference</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_web_service</td>
<td>name</td>
<td>server.xml</td>
</tr>
<tr>
<td>Name</td>
<td>cmdb_ci_web_application</td>
<td>name</td>
<td>serverindex.xml</td>
</tr>
<tr>
<td>App server</td>
<td>cmdb_ci_web_application</td>
<td>app_server</td>
<td>serverindex.xml</td>
</tr>
</tbody>
</table>

Relationships

These relationships are created in the CI Relationship (cmdb_rel_ci) table.

<table>
<thead>
<tr>
<th>Parent class</th>
<th>Relationship type</th>
<th>Child class</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_app_server_websphere</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_linux_server</td>
</tr>
<tr>
<td>cmdb_ci_win_server</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_win_server</td>
</tr>
<tr>
<td>cmdb_ci_app_server_websphere</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_web_application</td>
</tr>
<tr>
<td>cmdb_ci_web_service</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_app_server_websphere</td>
</tr>
<tr>
<td>cmdb_ci_websphere_cell</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_app_server_websphere</td>
</tr>
</tbody>
</table>

Database discovery

Discovery can find database applications, such as MySQL, Oracle, and MongoDB. Discovery can also find database management system software, such as MSSQL Server.

Database catalog discovery

The database catalog lists all the catalog objects, or databases, discovered for an instance of a database.

Databases vs. catalogs

Most database manufacturers use the term catalog to mean database, while Microsoft MSSQL uses the term database. Database catalogs can be imported into your instance from a third-party discovery tool, entered into the platform manually, or found by Discovery.

Important: Discovery does not support Oracle or MySQL database catalog discovery in the base system. You must create probes and sensors to find these. However, the Windows - MSSQL probe is available by default for MSSQL discovery. A pattern is also available. However, the pattern and the probes do not populate the same information.

To view a database catalog, navigate to Configuration > Database Catalogs and select a database.
**MSSQL usage of catalogs**

MSSQL database catalogs store metadata about an MSSQL database in the MSFT SQL Catalog (cmdb_ci_db_mssql_catalog) table. In the base system, the MSFT SQL Catalog form contains two fields: Name and Database instance. You can customize the form layout to show additional fields.

To access an MSFT SQL catalog record:

1. Navigate to Configuration > Database Instances > MSSQL and select a record.
2. In the instance record, select the MSFT SQL Catalogs related list.
3. Open an existing record from the list or click New to create a new catalog record.

**MSSQL server discovery**

Discovery identifies and classifies information about MSSQL servers on Windows machines.

A pattern performs the discovery of MSSQL instances. This pattern replaces the Windows - MSSQL probe for new instances. If you want to continue to use probes, you can deactivate the Horizontal discovery probe, which launches the pattern, and then activate the necessary probes.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

**Credentials**

Configure these credentials:

<table>
<thead>
<tr>
<th>Credential type</th>
<th>Additional notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Applicative credentials</strong></td>
<td>Only local credentials are supported. Domain users are not supported.</td>
</tr>
<tr>
<td><strong>Windows credentials</strong></td>
<td>The credentials need access to the following:</td>
</tr>
<tr>
<td></td>
<td>- A database user that can access these MSSQL tables:</td>
</tr>
<tr>
<td></td>
<td>sys.dm_exec_sessions and sys.dm_osSchedulers.</td>
</tr>
<tr>
<td></td>
<td>SERVERPROPERTY.</td>
</tr>
<tr>
<td></td>
<td>- The CI Type MSFT SQL Instance.</td>
</tr>
<tr>
<td>(optional)</td>
<td></td>
</tr>
<tr>
<td><strong>Applicative credentials</strong></td>
<td>The credentials need public access to the following:</td>
</tr>
<tr>
<td></td>
<td>- The target Windows host.</td>
</tr>
<tr>
<td></td>
<td>- The Microsoft SQL Server instance on the target Windows host. You must add the user to the SQL Server configuration.</td>
</tr>
<tr>
<td></td>
<td>- The MID Server host. The SMO libraries locally impersonate the credentials for the target system prior to connecting to the Microsoft SQL Server. This behavior is enforced by Active Directory. Authentication only succeeds if the domain requirements specified here are met.</td>
</tr>
</tbody>
</table>

**Note:** If there are no matching discovery credentials, probes may attempt to connect using the MID Server service credentials. The MID Server service credentials are only valid if the MID Server host is on same domain as the Microsoft SQL Server host.

Specify the CI type on the credential as MSFT SQL Instance.
Classifier, pattern, and probe

Discovery uses these classifiers, probes, and patterns for SQL discovery:

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft SQL Server</td>
<td>- Horizontal discovery probe:</td>
<td>MSSql DB On Windows Pattern</td>
</tr>
<tr>
<td></td>
<td>- Horizontal discovery probe:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Windows - MSSQL*</td>
<td></td>
</tr>
</tbody>
</table>

*Probes inactive on new instances.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

Data collected

The running processes of the database (the actual SQL server) is referred to as the database instance. The following data is collected in the MSFT SQL Instance (cmdb_ci_db_mssql_instance) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Instance</td>
<td>instance_name</td>
</tr>
<tr>
<td>TCP port</td>
<td>tcp_port</td>
</tr>
<tr>
<td>Port dynamic</td>
<td>port_dynamic</td>
</tr>
<tr>
<td>Version name</td>
<td>version_name</td>
</tr>
<tr>
<td>Edition</td>
<td>edition</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Engine edition</td>
<td>engine_edition</td>
</tr>
<tr>
<td>Install directory*</td>
<td>install_directory</td>
</tr>
<tr>
<td>CPU sockets*</td>
<td>cpu_sockets</td>
</tr>
<tr>
<td>Service Pack*</td>
<td>service_pack</td>
</tr>
</tbody>
</table>

*These fields are populated by the pattern only.

The database is referred to as the catalog. The following data is collected in the MSFT SQL Catalog (cmdb_ci_db_mssql_catalog) table:

**Note:** Only the Windows - MSSQL probe, not the MSSql DB On Windows pattern, populates this table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Status</td>
<td>status</td>
</tr>
</tbody>
</table>
Note: You can find the data for the actual server on which the MSSQL instance and catalogs resides in the Windows computer (cmdb_ci_win_server).

Discovery populates the MSSQL database (cmdb_ci_db_mssql_database) table with the MSSql DB on Windows Pattern:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Operational status</td>
<td>operational_status</td>
</tr>
</tbody>
</table>

For Microsoft SQL2000 servers, Discovery can match instance names to their process ID.

When Software Asset Management is activated, all discovered software is saved in at least two places in the CMDB: in the Software Installation (cmdb_sam_sw_install) table and the tables specific to the application type. A reference field is provided on these application-specific tables to the corresponding record in the Software Installation table.

Microsoft SQL Server cluster Discovery

Discovery identifies Microsoft SQL Server instances that are part of a cluster in the CMDB by the cluster name rather than as individual configuration items (CI). Discovery identifies Microsoft SQL Server instances in the CMDB that are part of a cluster by creating a Runs::Runs on relationship between the instance and its associated cluster, as well as the Runs::Runs on relationship between the instance and the actual host on which it runs.
Viewing discovered relationships

By default, a CI record for a server (node) hosting an SQL Server instance that is part of a cluster does not display the instance or the cluster as related items.

To see the cluster information on a CI record:

1. Add the **Cluster Nodes** related list to the form. This list displays the name of the node and the parent cluster.
### Configuration

<table>
<thead>
<tr>
<th>Name</th>
<th>cluster-node2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company</td>
<td></td>
</tr>
<tr>
<td>Serial number</td>
<td>VMware-42 1d e6 78 46 f6 a1 6b-0f 9c b7 e1</td>
</tr>
<tr>
<td>Model ID</td>
<td>VMware, Inc. VMware Virtual Platform</td>
</tr>
<tr>
<td>Assigned to</td>
<td></td>
</tr>
<tr>
<td>Default Gateway</td>
<td>10.11.144.1</td>
</tr>
</tbody>
</table>

### Related Links

- Discover now
- Subscribe
- Show Discovery events

<table>
<thead>
<tr>
<th>Cluster Nodes</th>
<th>Name</th>
<th>Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Server = cluster-node2*
2. Click the cluster name in the related list to drill into the CI record for that cluster. Individual SQL Server instances running on that cluster are listed in the Related Items hierarchy. The names in the list contain the instance name prepended to the host name with the @ symbol. In this example, MSSQLCLUSTER is the SQL Server instance name, and cluster-node2 is the name of the Windows host.
3. Click the name of the SQL Server instance in the Related Items list to open the record for that CI. The instance name is expressed in the format `instance@host name` and displays the relationship hierarchy to the configured number of levels, including the system databases that the instance uses, the cluster on which the instance runs, and the server on which the cluster runs.
<table>
<thead>
<tr>
<th><strong>Name</strong></th>
<th>MSSQLCLUSTER@cluster-node2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Version</strong></td>
<td>11.3.6020.0</td>
</tr>
<tr>
<td><strong>Configuration directory</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Running process</strong></td>
<td>sqlservice.exe</td>
</tr>
<tr>
<td><strong>TCP port(s)</strong></td>
<td>61433</td>
</tr>
<tr>
<td><strong>Configuration file</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Advanced**

<table>
<thead>
<tr>
<th><strong>Instance Name</strong></th>
<th>MSSQLCLUSTER</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Version name</strong></td>
<td>-- None --</td>
</tr>
<tr>
<td><strong>Edition</strong></td>
<td>Enterprise Edition (64-bit)</td>
</tr>
<tr>
<td><strong>Engine edition</strong></td>
<td>-- None --</td>
</tr>
<tr>
<td><strong>Service pack</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Related Items**

- **Contains** - MS SQL Databases
  - [master](#)
  - [model](#)
  - [msdb](#)
  - [msqlystemresource](#)
  - [tmpdb](#)

- **Runs on** - Windows Clusters
  - [devopsclu](#)

- **Runs on** - Windows Servers
  - [cluster-node2](#)
Defining a clusterable process

1. Open the desired process classifier (Discovery Definition > CI Classification > Process).
2. Add the Parameters related list (if not visible).
3. Click New in the Parameters related list and add a new parameter, using the following values:
   - **Name**: Unique name of your choosing.
   - **Type**: Enter `Cluster`.
   - **Value**: Enter the following statement.

```javascript
resourceType: "Resource Type",
isMatch: function(process, resource) {
   // javascript function that returns true if a resource matches the classified process
}
```

### Defining Clusterable Processes

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>resourceType</td>
<td>Resource type the platform should look for in the cluster's Windows Cluster Resources related list.</td>
</tr>
<tr>
<td>isMatch</td>
<td>Determines if the cluster resource that is found, based on the type, is a match for the classified process being examined.</td>
</tr>
</tbody>
</table>
MySQL discovery
Discovery can identify an instance of MySQL that is running on UNIX or Windows operating systems.

Requirements
Discovery searches for the MySQL configuration file location from the following areas:

- UNIX: Discovery searches for the MySQL configuration file location from the mysqld process, or port 3306.
- Windows: Discovery searches for the MySQL configuration file location from the mysqld.exe process, or port 3306.

For each process, the following process parameters are explored in the following order:

--defaults-extra-file
--defaults-file
If the MYSQL configuration file location is not found from that search, then the following occurs:

- UNIX: The configuration file location defaults to `/etc/my.cnf`.
- Windows: No default configuration file location exists, and the probe to read the configuration file location is skipped.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

### Credentials

Configure these credentials:

- **Windows credentials**
- **SSH credentials**
- (optional) **Applicative credentials**

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#) for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

### Classifiers, patterns, and probes

Discovery uses these classifiers, probes, and patterns for MySQL discovery:

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL Server</td>
<td>· Horizontal Pattern: launches patterns</td>
<td>MySQL server On Windows and Linux Patte</td>
</tr>
<tr>
<td></td>
<td>· Windows - MySQL Configuration*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Windows - MySQL Version*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· MySQL - Configuration*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· MySQL - Version*</td>
<td></td>
</tr>
</tbody>
</table>

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See [Add the Horizontal Pattern probe to a classifier](#) for instructions.

### Data collected

Discovery populates these fields in the MySQL Instance (cmdb_ci_db_mysql_instance) table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL configuration</td>
<td>myconf</td>
<td>my.cnf</td>
</tr>
<tr>
<td>TCP port(s)</td>
<td>tcp_port</td>
<td>running_process</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
<td>mysqlld</td>
</tr>
</tbody>
</table>
MongoDB discovery
Discovery creates or updates a CMDB record when it detects a running instance of MongoDB.

The following data is collected:

<table>
<thead>
<tr>
<th>Data Collected by Discovery on MongoDB instances</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Label</strong></td>
</tr>
<tr>
<td>Version</td>
</tr>
<tr>
<td>Mongo configuration</td>
</tr>
<tr>
<td>TCP port(s)</td>
</tr>
</tbody>
</table>

PostgreSQL discovery
Discovery can find running instances of PostgreSQL on Windows and Linux systems.

Credentials and other prerequisites
These credentials are required:

- **SSH credentials**
- (optional) **Applicative credentials**

For a list of privileged commands that you need for Discovery and Service Mapping, see Service Mapping commands requiring a privileged user for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

The user must have root-level access to the database to access the `postgresql.conf` file.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

Classifiers, patterns, and probes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Trigger probes</th>
<th>Patterns</th>
</tr>
</thead>
</table>
| PostgreSQL Instance | - Horizontal Pattern: launches patterns  
- PostgreSQL - Configuration* (add the `must_sudo` parameter to this probe)  
- PostgreSQL - Version* | PostgreSQL DB |

*For new instances, these probes are inactive on the classifier. Discovery uses patterns for discovery.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.
Data collected

<table>
<thead>
<tr>
<th>Label</th>
<th>Table Name</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cmdb_ci_db_postgresql_instance</td>
<td>name</td>
<td>PostgreSQL Instance@hostname</td>
</tr>
<tr>
<td>Data Directory</td>
<td>cmdb_ci_db_postgresql_instance</td>
<td>data_dir</td>
<td>running process</td>
</tr>
<tr>
<td>TCP port</td>
<td>cmdb_ci_db_postgresql_instance</td>
<td>tcp_port</td>
<td>running process</td>
</tr>
<tr>
<td>SQL Configuration</td>
<td>cmdb_ci_db_postgresql_instance</td>
<td>postgres_conf</td>
<td>data_directory/ postgresql.conf</td>
</tr>
<tr>
<td>Version</td>
<td>cmdb_ci_db_postgresql_instance</td>
<td>version</td>
<td>postmaster/postgres</td>
</tr>
</tbody>
</table>

Relationships

<table>
<thead>
<tr>
<th>Parent class</th>
<th>Relationship</th>
<th>Child class</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_db_postgresql_instance</td>
<td>Runs on:Runs</td>
<td>cmdb_ci_windows_server or cmdb_ci_linux_server</td>
</tr>
</tbody>
</table>

Oracle database discovery

Discovery can identify an Oracle database instance that is running on UNIX or Windows operating systems. It can also find Oracle clusterware, which runs Real Application Clusters (RAC).

Patterns perform the discovery of Oracle installations. These patterns replace several probes for new instances. If you want to continue to use probes, you can deactivate the Horizontal discovery probe, which launches the pattern, and then activate the necessary probes.

Discovery does not discover the Oracle databases themselves, which are referred to as database catalogs.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

Credentials

Configure these credentials:
### Credential type | Additional information
--- | ---
**Applicative credentials** | Applicative credentials are only mandatory for two use cases:
- Creating connections to database links
- Software Asset Management activated

You must add the database user you configured during Oracle database installation. Specify Oracle instance `[cmdb_ci_db_ora_instance]` as the CI type.

The user must have access to these Oracle tables:
- V$option
- V$PARAMETER
- dba_hist_active_sess_history
- dba_users

The user must also have access to the `tnsnames.ora` parameters file.

**Windows credentials** | You can also use Windows and SSH credentials in addition to applicative credentials. For Windows credentials, the user must have read permissions to the `tnsnames.ora` file. SSH credentials, must allow read permission to the `oratab` file and read permission to the System Parameter file.

For a list of privileged commands that you need for Discovery and Service Mapping, see [Service Mapping commands requiring a privileged user](#).

For a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**SSH credentials** | The following commands are run for Oracle clusterware:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`ps -ef</td>
<td>grep 'crsd.bin'</td>
</tr>
<tr>
<td>`ps -ef</td>
<td>grep LISTENER</td>
</tr>
<tr>
<td>`/u01/app/12.1.0.2/grid/bin/olsnodes -c</td>
<td>egrep -v 'error</td>
</tr>
</tbody>
</table>

### Additional requirements

Discovery identifies a running instance of an Oracle database on UNIX from the process that starts with `ora_pmon_`. Ensure this process is running in the IP range you designate for Discovery.

Discovery identifies a running instance of an Oracle database on Windows from the `oracle.exe` process. Ensure this process is running in the IP range you designate.

For Oracle clusterware, you also need permissions to read the `/etc/hosts/` file, which associates host names with IP addresses.

The following commands are run for Oracle clusterware:
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/crsctl query crs softwareversion</code></td>
<td>Gets Oracle CRS cluster ID.</td>
</tr>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/crsctl query crs softwareversion</code></td>
<td>Gets Oracle CRS version.</td>
</tr>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/olsnodes</code></td>
<td>Gets Oracle CRS nodes.</td>
</tr>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/crsctl status server</code></td>
<td>Gets Oracle CRS node status.</td>
</tr>
<tr>
<td>`/u01/app/12.1.0.2/grid/bin/crs_stat -f</td>
<td>tr '\n' '#'`</td>
</tr>
<tr>
<td>`/u01/app/12.1.0.2/grid/bin/crsctl stat res -p</td>
<td>grep -v CHECK_R</td>
</tr>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/srvctl config scan</code></td>
<td>Gets Oracle RAC SCAN VIP configurations.</td>
</tr>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/srvctl config vip -node ol7-121-rac1</code></td>
<td>Gets Oracle CRS VIPs name.</td>
</tr>
<tr>
<td><code>/u01/app/12.1.0.2/grid/bin/srvctl config vip -node ol7-121-rac1</code></td>
<td>Get Oracle CRS VIPs ip_address.</td>
</tr>
</tbody>
</table>

Sudo permission to run: `sudo /u01/app/12.1.0.2/grid/bin/ocrcheck | egrep -v 'error|return code'`

**Classifiers, patterns, and probes**

Discovery uses these classifiers, probes, and patterns for Oracle database discovery:
Oracle Instance

- Horizontal discovery probe: launches patterns
- Powershell-Oracle - Instance PFile (for Windows)*
- Powershell-Oracle - Instance Version (for Windows)*
- SSHCommand-Oracle - Instance Details (for Linux)*

Application patterns:
- Oracle DB On Windows Pattern
- Oracle DB on UNIX Pattern

This shared library pattern used by the Linux Server pattern:
- UNIX Cluster - ORACLE Clusterware

Note: This shared library pattern is triggered if the crsd.bin process is running on the Oracle database server.

Oracle Listener

- SSHCommand-Oracle - Listener Details

none

*Probes inactive for new instances.

To use patterns, verify that the correct pattern is specified in the horizontal pattern probe on the classifier. See Add the Horizontal Pattern probe to a classifier for instructions.

Data collected

The following data is collected on the Oracle Instances (cmdb_ci_db_ora_instance) table for both UNIX and Windows:

Data collected on UNIX machines

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SID</td>
<td>sid</td>
<td>Obtained from the name of the process that starts with ora_pmon_</td>
</tr>
<tr>
<td>Install directory</td>
<td>install_directory</td>
<td>Obtained from the path of ORA_HOME</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
<td>Obtained in this order:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- From the output of the ORA_HOME/bin/sqlplus /NOLOG command</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- From the output of the ORA_HOME/bin/lsnrctl status command</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- From the path of ORA_HOME</td>
</tr>
<tr>
<td>Label</td>
<td>Field name</td>
<td>Additional information</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>Parameter file</td>
<td>pfile</td>
<td>The following locations are explored for the location of the System Parameter File. If this file does not exist in one of the explored locations, Discovery does not find the file and reports an error.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <code>oracle_home/dbs/spfileSID.ora</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <code>oracle_home/dbs/spfile.ora</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <code>oracle_home/dbs/initSID.ora</code></td>
</tr>
<tr>
<td>Oracle Home</td>
<td>oracle_home</td>
<td>Obtained from the <code>ORATAB</code> file.</td>
</tr>
<tr>
<td>Edition</td>
<td>edition</td>
<td></td>
</tr>
</tbody>
</table>

**Data collected on Windows machines**

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SID</td>
<td>sid</td>
<td>Obtained from the process parameter that is passed to the <code>oracle.exe</code> process.</td>
</tr>
<tr>
<td>Install directory</td>
<td>install_directory</td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
<td>Obtained from the output of the <code>ORA_HOME/bin/sqlplus.exe -V</code> command.</td>
</tr>
<tr>
<td>Parameter file</td>
<td>pfile</td>
<td>The following locations are explored for the location of the System Parameter File. If this file does not exist in one of the explored locations, Discovery does not find the file and reports an error.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <code>oracle_home\database\spfileSID.ora</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <code>oracle_home\database\spfile.ora</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <code>oracle_home\database\initSID.ora</code></td>
</tr>
<tr>
<td>Oracle Home</td>
<td>oracle_home</td>
<td>Parsed from the path of <code>oracle.exe</code></td>
</tr>
<tr>
<td>Edition</td>
<td>edition</td>
<td></td>
</tr>
</tbody>
</table>

**Oracle options**

Discovery can also find the Oracle options that you enable on the instance. See the Oracle website for more information on options [https://docs.oracle.com/en/](https://docs.oracle.com/en/) Discovery saves these to the Oracle Options (samp_oracle_options) table.
<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance</td>
<td>instance</td>
</tr>
<tr>
<td>Option</td>
<td>option</td>
</tr>
</tbody>
</table>

Data collected on clusterware

<table>
<thead>
<tr>
<th>Table</th>
<th>Labels and field names</th>
</tr>
</thead>
</table>
| Unix Cluster (cmdb_ci_unix_cluster) | Name (name)  
Cluster status (cluster_status)  
Cluster type (cluster_type)  
IP address (ip_address)  
Short description (short_description)  
Cluster ID (cluster_id)  
Cluster version (cluster_version) |
| cmdb_ci_unix_cluster_node | Name (name)  
Node state (node_state)  
IP address (ip_address)  
Node status (node_status) |
| cmdb_ci_unix_cluster_resource | Name (name)  
Resource type (resource_type)  
Resource status (resource_status)  
Properties (properties) |
| cmdb_ci_cluster_vip | Name (name)  
IP address (ip_address)  
Cluster name (cluster_name)  
Cluster ID (cluster_id) |

On the dependency view, a +1 on an Oracle database server signifies that it is part of a cluster.

**Oracle instances on virtual machines**

When Oracle instances are hosted on virtual machines, the system creates relationships between CIs.

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle instance</td>
<td>Hosted on</td>
<td>Computer (cmdb_ci_computer) or Server (cmdb_ci_server)</td>
</tr>
</tbody>
</table>

**Note:** This is a virtual machine. The Is virtual field is true.
Oracle instances on virtual machines that are hosted on AWS or Azure clouds

<table>
<thead>
<tr>
<th>Base class</th>
<th>Relationship</th>
<th>Dependent class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle instance</td>
<td>Hosted on</td>
<td>Computer (cmdb_ci_computer)</td>
</tr>
<tr>
<td>Computer (cmdb_ci_computer)</td>
<td>Virtualized by</td>
<td>Computer (cmdb_ci_computer)</td>
</tr>
<tr>
<td>Computer (cmdb_ci_computer)</td>
<td>Virtualized by</td>
<td>Virtual Machine Instance (cmdb_ci_vm_instance)</td>
</tr>
<tr>
<td>Virtual Machine Instance (cmdb_ci_vm_instance)</td>
<td>Hosted on</td>
<td>AWS Datacenter (cmdb_ci_aws_datacenter) or Azure Datacenter (cmdb_ci_azure_datacenter)</td>
</tr>
</tbody>
</table>

Relationships for clusterware

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unix Cluster (cmdb_ci_unix_cluster)</td>
<td>Virtualizes::Virtualized by</td>
<td>Cluster Virtual IP (cmdb_ci_cluster_vip)</td>
</tr>
<tr>
<td>Unix Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
<td>Defines resources for::Gets resources from</td>
<td>Unix Cluster Node (cmdb_ci_unix_cluster_node)</td>
</tr>
<tr>
<td>Cluster Virtual IP (cmdb_ci_cluster_vip)</td>
<td>Uses::Used by</td>
<td>Unix Cluster Node (cmdb_ci_unix_cluster_node)</td>
</tr>
<tr>
<td>Unix Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>Cluster of::Cluster</td>
<td>Unix Cluster (cmdb_ci_unix_cluster)</td>
</tr>
</tbody>
</table>
### SAP HANA Database discovery

Discovery creates or updates a CMDB record when it detects a running instance of SAP HANA database.

By default, Discovery uses the **SAP HANA DB** pattern to discover the SAP HANA Database.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article **KB0694477** for more information.

### Data collected by Discovery for SAP HANA Database

The following data is collected in the HANA Database `[cmdb_ci_appl_sap_hana_db]` table:

<table>
<thead>
<tr>
<th>Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_address</td>
</tr>
<tr>
<td>Class</td>
<td>sys_class_name</td>
</tr>
<tr>
<td>Fully qualified domain name</td>
<td>fqdn</td>
</tr>
<tr>
<td>Type</td>
<td>type</td>
</tr>
<tr>
<td>Version</td>
<td>version</td>
</tr>
<tr>
<td>Installation directory</td>
<td>install_directory</td>
</tr>
</tbody>
</table>

### Sybase discovery

Discovery creates or updates a CMDB record when it detects an instance of Sybase.

By default, Discovery uses the **Sybase** pattern to discover Sybase.

For a list of privileged commands that you need for Discovery and Service Mapping, see **Service Mapping commands requiring a privileged user** for a list of the commands that require elevated rights to discover and map Unix-based hosts in your organization.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article **KB0694477** for more information.

### Data collected by Discovery for Sybase

The following data is collected in the Sybase Instance `[cmdb_ci_db_syb_instance]` table:
Storage discovery

Discovery collects information on Direct Attached Storage (DAS), Storage Area Networks (SAN), and Network Attached Storage (NAS).

Storage can be located on specialized devices, such as Storage Arrays, Fibre Channel Switches, iSCSI disks, or on host operating systems, including Windows, Linux, and Solaris.

Discovery finds and maps dependencies for the following types of storage:

- Direct-attached storage (DAS), network-attached storage (NAS), or storage area network (SAN).
- NAS or SAN storage that is discovered via a Storage Management Initiative Specification (SMI-S) and Common Information Model (CIM).
- Virtual storage for VMware ESX servers and Linux Kernel-based Virtual Machines (KVM).

Discovery maps this storage to the underlying physical storage.

Discovery of storage via a host reconciles data and creates relationships between the host's file systems and associated local storage devices. The local storage devices represent the storage available to the host, whether it's directly attached or provided by Fibre Channel or iSCSI. This reconciliation assumes that the storage server has been discovered first.

Note: Tape storage drivers are not discovered.

Discovery collects and creates CIs in the CMDB for the following information:

- File systems (local and NAS).
- Disks (both SAN disks and DAS drives).
- Fibre Channel (FC) HBAs and ports.
- Linux Volume Manager (LVM) volumes. LVM volume data resides in the Storage Pool (cmdb_ci_storage_pool) table.

- **Veritas Volume Manager** disks, subdisks, disk groups, plexes, and volumes.

Note: For details about the discovery of direct attached or multipath block storage provisioned on a Linux host, see [KB0622583](#).

Probes, sensors, and patterns

- KVM - Storage Pools: identifies storage attached to KVM virtual machines.
- Linux - Storage: identifies storage attached to systems running the Linux operating system.
- Solaris - Storage: identifies storage attached to systems running the Solaris operating system.

- **Windows - Storage 2012**: identifies storage attached to systems running Windows 2012 and later.
  - **Windows - Storage 2012 - PS**: identifies storage attached to Windows systems, using PowerShell.
  - **Windows - Storage 2012 - WMI**: identifies storage attached to Windows systems, using WMI Runner.

- **VMWare - vCenter ESX Hosts Storage**: collects information about ESX servers and creates relationships from datastores to underlying disks.

- **Patterns for NetApp storage discovery**:
  - **NetApp 7-mode**: finds NetApp servers via REST with two nodes.
  - **NetApp cluster mode**: finds NetApp servers via REST when more than two nodes are connected through a cluster interconnect switch.

**Note**: Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

**Requirements**

**Windows**

- For Windows, use one of the following supported host configurations:
  - Windows Server 2012, DAS or NAS with Fibre Channel (FC) or Internet Small Computer System Interface (iSCSI).
  - Windows Server 2008, DAS or NAS with FC or iSCSI.

- Install the `fcinfo.exe` tool on Windows 2008 and 2012 servers that attach to storage via FC.
- Install Windows Management Instrumentation (WMI).
- Enable Powershell on the MID Server host server.
- Provide the instance with the necessary credentials to the host server.
- Put the MID Server and the target machine in the same domain or add the target machine to the trusted host list on the MID Server machine.

**Linux**

- For Linux, use one of the following supported host configurations:
  - Solaris, DAS, NAS, or SAN with iSCSI
  - CentOS, DAS, NAS, or SAN with FC or iSCSI
  - Ubuntu Server, DAS, NAS, or SAN with iSCSI

- Provide the device with SSH credentials that have root or sudo access.
• Provide the MID Server with the necessary credentials to the host server.

Relationships

Discovery creates the following relationships for storage CIs:

<table>
<thead>
<tr>
<th>Parent Component</th>
<th>Relationship</th>
<th>Child Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Export (cmdb_ci_storage_export)</td>
<td>Exports to::Imports from</td>
<td>Storage Device (cmdb_ci_storage_device)</td>
</tr>
<tr>
<td>Fibre Channel Disk (cmdb_ci_fc_disk)</td>
<td>Provides::Provided by</td>
<td>File System (cmdb_ci_file_system)</td>
</tr>
<tr>
<td>ISCSI Disk (cmdb_ci_iscsi_disk)</td>
<td>Provides::Provided by</td>
<td>File System (cmdb_ci_file_system)</td>
</tr>
<tr>
<td>NAS File System (cmdb_ci_nas_file_system)</td>
<td>Allocated from::Allocated to</td>
<td>Storage File Share (cmdb_ci_storage_fileshare)</td>
</tr>
</tbody>
</table>

Configure discovery of hosts with attached storage
Discovery can find information about UNIX, Linux, and Windows hosts with attached storage.

admin or discovery_admin

Configure Discovery for UNIX or Linux hosts with attached storage
Discovery can find information about a UNIX or Linux host and attached storage.

1. Optional: On the host, assign elevated privileges to the account that Discovery will use.
2. On the ServiceNow instance, add credentials to the Discovery Credentials table.
3. Create a Discovery Schedule for each host IP address.
4. Optional: Create a behavior that uses a functionality definition with a wbem port probe to make the initial port-scanning phase (Shazzam) more efficient.
5. Run network discovery.

Configure Discovery for Windows hosts with attached storage
Discovery can find information about a Windows host and attached storage.

1. For Windows 2008 servers that attach to storage via FC. For servers that are using newer operating systems, similar functionality is present with the operating system; this step is not required:
   b) Set the environment path for the fcinfo executable and run the fcinfo.exe file.
2. For all Windows hosts including Windows 2008, add Windows credentials to the Discovery Credentials table.
3. On the ServiceNow instance, create a Discovery Schedule for each host IP address.
4. Optional: Create a behavior that uses a functionality definition with a wbem port probe to make the initial port-scanning phase (Shazzam) more efficient.
5. Run network discovery.
Note: If the host also connects to a NAS or SAN storage array, set up the SMI-S Provider and CIM credentials.

Discovery data collected for storage via a host
Discovery gathers information about storage units that connect to Linux, Solaris, and Windows hosts via a local I/O port or Host Bus Adapter (HBA).

Probes used to discover storage from a host

<table>
<thead>
<tr>
<th>Table</th>
<th>Probes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk (cmdb_ci_disk)</td>
<td>• KVM</td>
</tr>
<tr>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Disk Partition (cmdb_ci_disk_partition)</td>
<td>• KVM</td>
</tr>
<tr>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Fibre Channel Disk (cmdb_ci_fc_disk)</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td>File System (cmdb_ci_file_system)</td>
<td>• KVM</td>
</tr>
<tr>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>iSCSI Disk (cmdb_ci_iscsi_disk)</td>
<td>• KVM</td>
</tr>
<tr>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>NAS File System (cmdb_ci_nas_file_system)</td>
<td>• KVM</td>
</tr>
<tr>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
</tbody>
</table>
Table and column | Reference | Target data element
--- | --- | ---
cmdb_ci_storage_device.computer | refers to the | cmdb_ci_computer
cmdb_ci_storage_device.provided_by | refers to the | cmdb_ci_fc_port (for FC only)
cmdb_ci_disk_partition.disk | is a partition of | cmdb_ci_disk
cmdb_ci_storage_hba.computer | is the | cmdb_ci_computer
cmdb_ci_fc_port.controller | contains the | cmdb_ci_storage_hba
cmdb_ci_fc_port.computer | is the same | cmdb_ci_computer as cmdb_ci_storage_hba.computer
cmdb_ci_storage_volume.computer | is the | cmdb_ci_computer
cmdb_ci_storage_volume.provided_by | is the | cmdb_ci_storage_pool or cmdb_ci_storage_pool or cmdb_ci_storage_device (providing storage)
cmdb_ci_storage_pool.hosted_by | is the | cmdb_ci_computer
cmdb_ci_computer | the pool is on | cmdb_ci_storage_pool.container
cmdb_ci_storage_pool.container | is the | cmdb_ci_storage_pool or cmdb_ci_storage_pool or cmdb_ci_storage_device containing the pool (if the pool is present)
cmdb_ci_storage_pool_member.pool | is the | cmdb_ci_storage_pool
cmdb_ci_storage_pool_member.storage | is the | cmdb_ci_storage_pool, cmdb_ci_disk_partition or cmdb_ci_storage_device providing storage

Data collected for HBAs on Linux and ESX servers

Discovery collects information on host bus adapters (HBA) for fiber channel enabled devices connected to Linux and ESX servers.
### HBA tables and fields

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>name</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>WWNN</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>wwnn</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>Model ID</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>model_id</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>Computer</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>computer</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
</tbody>
</table>

### Fibre Channel ports

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>WWPN</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>wwpn</td>
<td>VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>WWNN</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>wwnn</td>
<td>VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>Port type</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>port_type</td>
<td>VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>Speed</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>speed</td>
<td>VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>Controller</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>controller</td>
<td>VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td>Label</td>
<td>Table</td>
<td>Field name</td>
<td>Source</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------</td>
<td>------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>Computer</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>computer</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Linux - Storage</td>
</tr>
</tbody>
</table>

### fibre channel targets

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC Disk</td>
<td>Fibre Channel Targets (cmdb_fc_target)</td>
<td>fc_disk</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Created by</td>
<td>Fibre Channel Targets (cmdb_fc_target)</td>
<td>sys_created_by</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Created</td>
<td>Fibre Channel Targets (cmdb_fc_target)</td>
<td>sys_created_on</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>SysID</td>
<td>Fibre Channel Targets (cmdb_fc_target)</td>
<td>sys_id</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Updates</td>
<td>Fibre Channel Targets (cmdb_fc_target)</td>
<td>sys_mod_count</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Updated by</td>
<td>Fibre Channel Targets (cmdb_fc_target)</td>
<td>sys_updated_by</td>
<td>• Linux - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Solaris - Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Label</td>
<td>Table</td>
<td>Field name</td>
<td>Source</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------</td>
<td>------------</td>
<td>---------------------------------------------</td>
</tr>
</tbody>
</table>
| Updated | Fibre Channel Targets (cmdb_fc_target)     | sys_updated_on | • Linux - Storage  
|         |                                            |            | • Solaris - Storage  
|         |                                            |            | • Windows - Storage 2008  
|         |                                            |            | • Windows - Storage 2012  |
| WWNN    | Fibre Channel Targets (cmdb_fc_target)     | wwnn       | • Linux - Storage  
|         |                                            |            | • Solaris - Storage  
|         |                                            |            | • Windows - Storage 2008  
|         |                                            |            | • Windows - Storage 2012  |
| WWPN    | Fibre Channel Targets (cmdb_fc_target)     | wwpn       | • Linux - Storage  
|         |                                            |            | • Solaris - Storage  
|         |                                            |            | • Windows - Storage 2008  
|         |                                            |            | • Windows - Storage 2012  |

Data collected for HBAs on Solaris servers

Discovery on Solaris supports HBAs manufactured by:

- Emulex
- QLogic

Discovery populates these fields in the Storage HBA (cmdb_ci_storage_hba) table, using the Solaris - Storage probe and sensor.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Computer</td>
<td>computer</td>
</tr>
<tr>
<td>Device ID</td>
<td>device_id</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>manufacturer</td>
</tr>
<tr>
<td>Model ID</td>
<td>model_id</td>
</tr>
<tr>
<td>Serial number</td>
<td>serial_number</td>
</tr>
</tbody>
</table>

Discovery populates these fields in the Fibre Channel Port (cmdb_ci_fc_port) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Field name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Computer</td>
<td>computer</td>
</tr>
<tr>
<td>Controller</td>
<td>controller</td>
</tr>
<tr>
<td>WWPN</td>
<td>wwpn</td>
</tr>
<tr>
<td>Label</td>
<td>Field name</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>WWNN</td>
<td>wwnn</td>
</tr>
<tr>
<td>Port type</td>
<td>port_type</td>
</tr>
<tr>
<td>Speed</td>
<td>speed</td>
</tr>
<tr>
<td>Operational status</td>
<td>operational_status</td>
</tr>
</tbody>
</table>

Discovery populates these fields in the Fibre Channel Disk (cmdb_ci_fc_disk) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name</td>
</tr>
<tr>
<td>Storage type</td>
<td>storage_type</td>
</tr>
<tr>
<td>Device interface</td>
<td>device_interface</td>
</tr>
<tr>
<td>Device LUN</td>
<td>device_lun</td>
</tr>
</tbody>
</table>

Discovery populates these fields in the Fibre Channel Targets (cmdb_fc_target) table.

<table>
<thead>
<tr>
<th>Label</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC Disk</td>
<td>fc_disk</td>
</tr>
<tr>
<td>WWNN</td>
<td>wwnn</td>
</tr>
<tr>
<td>WWPN</td>
<td>wwpn</td>
</tr>
</tbody>
</table>

**Data collected for HBAs on Windows servers**

Discovery on Windows supports HBAs on any operating system.

**HBA tables and fields**

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>name</td>
<td>• VMware - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>WWNN</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>wwnn</td>
<td>• VMware - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Label</td>
<td>Table</td>
<td>Field Name</td>
<td>Source</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------------------------------------------</td>
<td>------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Model ID</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>model_id</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Computer</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>computer</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
</tbody>
</table>

**Fibre Channel ports**

<table>
<thead>
<tr>
<th>Label</th>
<th>Table</th>
<th>Field Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>WWPN</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>wwpn</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>WWNN</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>wwnn</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Port type</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>port_type</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
<tr>
<td>Speed</td>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>speed</td>
<td>• VMWare - vCenter ESX Hosts Storage probe</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Windows - Storage 2012</td>
</tr>
</tbody>
</table>
### Table schema

#### VxVM tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Extends table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Veritas Disk (cmdb_ci_veritas_disk)</td>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
</tr>
<tr>
<td>Veritas Disk Group (cmdb_ci_veritas_disk_group)</td>
<td>Configuration Item (cmdb_ci)</td>
</tr>
<tr>
<td>Veritas Plex (cmdb_ci_veritas_plex)</td>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
</tr>
<tr>
<td>Veritas Subdisk (cmdb_ci_veritas_subdisk)</td>
<td>Storage Pool Member (cmdb_ci_storage_pool_member)</td>
</tr>
<tr>
<td>Veritas Volume (cmdb_ci_veritas_volume)</td>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
</tr>
</tbody>
</table>

#### Data collected

#### VxVM data collected

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Veritas Disk (cmdb_ci_veritas_disk)</td>
<td>• name</td>
<td>Provided by::Provides relationship to the physical disk.</td>
</tr>
<tr>
<td></td>
<td>• size_bytes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• disk_group</td>
<td></td>
</tr>
</tbody>
</table>

---

**Data collected for Veritas Volume Manager on Linux**

Discovery collects disk and volume information for Veritas Volume Manager (VxVM) on Linux hosts and maps file systems mounted on Veritas volumes to the upstream storage provider.

**Note:** Ensure that VxVM is correctly installed and configured. If you are using 3rd party drivers, you must configure sudo permission for `vxdmpadm`.
Displaying data

VxVM Discovery maps file systems to the disks that supply storage. By default this is the only information displayed in the UI. The file system to disk relationship is shown as an upstream Stored on relationship on the File System form. To see the remainder of the information that Discovery captures for VxVM, you must add the appropriate related list to the Linux Server form. The available lists are:

- Veritas Disk Group
- Veritas Disk
- Veritas Plex
- Veritas Subdisk (Computer)
- Veritas Subdisk (Storage)
- Veritas Volume

**Storage Discovery via SMI-S and CIM**

Discovery can explore storage devices that contain a Storage Management Initiative Specification (SMI-S) provider that is a specialized Common Information Model (CIM) server.

To see the current list of vendors and products conforming to SMI-S as tested by SNIA, see the [SNIA website](https://www.snia.org/).

Other types of storage, such as storage that is attached via a host, VM storage that is available on VMware ESX servers, and storage for Kernel VM (KVM), use a separate set of probes and sensors.
Discovery collects information about storage area networks (SAN) and network-attached storage (NAS) from specialized devices, such as storage arrays and Fibre Channel (FC) switches. Discovery collects and creates CIs in the CMDB for the following information items:

- Array disks, pools, and volumes
- Fibre Channel HBAs, ports, and controllers
- FC exports
- Fabrics, including endpoints, zoning, and switches
- Dependencies between storage sub-components

**Requirements**

- A CIM server using SMI-S 1.5 or later.
- NAS and SAN systems from major vendors such as EMC, Hitachi, HP, and NetApp. SAN storage devices must use FC.
- FC switches from major vendors such as Brocade and Cisco.
- The CIM credentials must be available for SMI-S configuration. The CIM credentials can be different than the credentials for the system hosting the CIM server.
- Ensure that SMI-S is enabled for your storage product.

**Note:** Because the SMI-S Provider caches storage device information, the Discovery query to the provider does not affect storage device performance.
SMI-S Discovery architecture

![Diagram of SMI-S Discovery architecture]

CIM architecture

CIM probes can explore any device based on the Common Information Model (CIM) by querying a CIM server, also referred to as a CIMOM - Common Information Model Object Manager. By default, Discovery uses CIM probes to explore storage systems as well as to get the serial numbers of ESX servers.

Discovery queries SMI-S compliant storage devices using CIMIQL queries.

The following components are part of CIM:

- **Common Information Model (CIM):** CIM allows multiple parties to exchange information about managed elements. CIM represents these managed elements and the management information, while providing the mechanism to actively control and manage the elements.
- **Storage Management Initiative Specification (SMI-S):** SMI-S is a standard of use that describes methods for storage discovery on the vendor’s side. ServiceNow uses SMI-S to determine how to discover CIM. SMI-S is based on the Common Information Model (CIM) and the Web-Based
Enterprise Management ( WBEM ) standards, which define management functionality via HTTP. The main objective of SMI-S is to enable management of dissimilar storage products. ServiceNow supports SMI-S version 1.5 or higher.

![CIM SMI-S Standard Diagram](image)

**CIM SMI-S Standard Diagram**

- **Web-Based Enterprise Management ( WBEM )**: WBEM defines a particular implementation of CIM, including protocols for discovering and accessing each CIM implementation.
- **Service Location Protocol ( SLP )**: SLP is an ad hoc protocol for retrieving and associating configuration information about CIM servers, such as default paths, capabilities, and the exact interop namespace. Discovery retrieves the interop namespace of a CIM server via SLP and passes that information to the CIM Classify probe. SLP, referred to here as the SLP server, uses service agents ( SA ) to gather and disseminate information about a CIM server on a subnet. A subnet can have multiple service agents.

**Note:**

The `mid.cim.interop.namespace` system property defines four default storage namespaces:
- interop
- root/interop
- root/pg_interop
- pg_interop

If you are using multiple storage vendors with custom namespaces not specified as one of the defaults, add the new namespaces to the comma-separated list in this property.
If you intend to continue using any of the default namespaces, make sure to include them in the property.

SLP and WBEM support

SLP is required for CIM Discovery as it is part of the Storage Management Initiative Specification (SMI-S) stack. Some storage devices may support the WBEM protocol, but may not support SLP.

You can manually register the WBEM services on SLP using a common Linux tool like `slptool`. This tool has a command line interface that you can use to make SLPv2 User Agent (UA) requests, which usually come with the SLP daemon package. To register a service, provide a URL and list of attributes. An example can be extracted from a working SLP server by using the same tool.

Storage Discovery table schema

This diagram displays the disk hierarchical schema for storage Discovery.
Disk hierarchical schema

Storage Discovery tables and probes

Discovery uses the following tables and probes to gather information about storage devices that are managed by a SMI-S provider.

Data Collected

<table>
<thead>
<tr>
<th>Table</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk (cmdb_ci_disk)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Fibre Channel Export (cmdb_ci_fc_export)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Fibre Channel Port (cmdb_ci_fc_port)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>SAN Export (cmdb_ci_san_export)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Storage Controller (cmdb_ci_storage_controller)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Table</td>
<td>Probe</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>Storage Export (cmdb_ci_storage_export)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Storage File Share (cmdb_ci_storage_fileshare)</td>
<td>SMI - Array - File Shares</td>
</tr>
<tr>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Storage Server (cmdb_ci_storage_server)</td>
<td>CIM - Identity</td>
</tr>
<tr>
<td>Storage Switch (cmdb_ci_storage_switch)</td>
<td>CIM - Identity</td>
</tr>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Storage Device (cmdb_ci_storage_device)</td>
<td>SMI 1.5 - Storage Server</td>
</tr>
<tr>
<td>Storage Area Network (cmdb_ci_san)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Connection (cmdb_ci_san_connection)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Endpoint (cmdb_ci_san_endpoint)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Fabric (cmdb_ci_san_fabric)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Zone (cmdb_ci_san_zone)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Zone Alias (cmdb_ci_san_zone_alias)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Zone Alias Member (cmdb_ci_san_zone_alias_member)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Zone Member (cmdb_ci_san_zone_member)</td>
<td>SMI - Fabric</td>
</tr>
<tr>
<td>SAN Zone Set (cmdb_ci_san_zone_set)</td>
<td>SMI - Fabric</td>
</tr>
</tbody>
</table>

These probes are available but not currently used:
- SMI - Storage Server
- SMI - Array - Controllers
- SMI - Array - Disks
- SMI - Array - Pools
- SMI - Array - Ports
- SMI - Array - Volumes

Discovery of storage area networks (SAN)
Discovery collects information about storage area networks from specialized devices, such as storage arrays and Fibre Channel (FC) switches, and creates specific references between the tables in the SAN schema.

**ServiceNow SAN schema**
This diagram shows the tables in the SAN schema and the default references defined between them.
Multipath SAN storage example

This diagram shows the relationship of SAN storage volumes to a host computer. Storage accessible to the Linux host consists of two physical fibre channel (FC) disks, mpatha and mpathb, connected to the host via a multipath FC SAN. The two HBAs on the host interface are connected with two fibre cables each to separate FC switches for failover capability. The FC fabric switches are connected to two FC storage processors on the storage server. In this example, each storage processor is connected to the storage volume's, **LUN 1** and **LUN 2**.

How CIM Discovery works

This is the processing flow for classifying Common Information Model (CIM) storage systems.
Processing flow

1. The Shazzam probe launches the wbem port probe as part of network discovery.
2. The wbem port probe detects activity on target ports SLP 427, CIM 5989 and 5988, and then examines the Service Registry Queries related list, at Discovery Definition > Port Probes, for the SLP query. The base system provides this query is provided to detect the service:wbem service type, which indicates the presence of an SLP server.
3. The Shazzam probe launches a scanner for the WBEM service type. The scanner retrieves:
   - The attributes of the service from the SLP server.
   - The interop namespaces of CIM servers in the network.
4. The scanner appends the namespace values it finds to the port probe results.
5. The wbem port probe appends the SLP data it carries to the CIM Classify probes.
6. The CIM Classify probe uses that information to explore the CIM servers.

CIM Shazzam Processing Diagram

The wbem port Probe

The wbem probe stores the data it retrieves in the CIM Classification [discovery_classy_cim] table. To view the wbem port probe, navigate to Discovery Definition > Port Probes.
SLP query

The SLP query detects the wbem service (service:wbem) on an SLP server and gathers the attributes of the service. To view the SLP Query record, open the wbem port probe record and select SLP Query from the Service Registry Queries related list.

SLP Query record

CIM - Classify probe

The wbem port probe appends the SLP data it carries, including namespaces, to the CIM - Classify probe before launching it. The CIM classification probe extracts VMware ESX serial numbers and connector relationships between the SAN and NAS components from CIM Servers in the network.

To access the CIM classification probe, navigate to Discovery Definition > Probes and select CIM - Classify from the list of probes.
Note:
The **mid.cim.interop.namespace** system property defines four default storage namespaces:
- interop
- root/interop
- root/pg_interop
- pg_interop

If you are using multiple storage vendors with custom namespaces not specified as one of the defaults, add the new namespaces to the comma-separated list in this property. If you intend to continue using any of the default namespaces, make sure to include them in the property.
### CIM Probe

**CIM - Classify**

**Name**: CIM - Classify

**Class**: CIM Probe

**Description**: CIM Classification Object Manager (CIMOM) probes, queries are not provided directly. Rather, they are aggregated from all active CIM Object Manager Classification Criteria.

Queries are dynamically built by collecting all CIM CI Classifications and aggregating their Classification Criteria. Duplicate CIM queries are merged.

CIM Queries are dynamically created by merging all CIM CI Classification Criteria.

#### Related Links

**Test probe**

<table>
<thead>
<tr>
<th>CIM Queries</th>
<th>Probe parameters</th>
<th>Sensors (1)</th>
<th>Included by MultiProbe</th>
<th>Responds to Probes</th>
<th>Discovery Port Probes (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Probe parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Value script</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>quals</td>
<td>// Aggregates CIM statements for all clas...</td>
<td>true</td>
<td></td>
</tr>
<tr>
<td>statement_names</td>
<td>// Aggregates CIM statement names for all...</td>
<td>true</td>
<td></td>
</tr>
</tbody>
</table>

**Actions on selected rows...**

1 to 2 of 2

---

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.

960
## SMI-S and CIM probes and sensors

<table>
<thead>
<tr>
<th>Probe/Sensor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIM - Identity</td>
<td>Identifies a system via CIM per SMI-S.</td>
</tr>
<tr>
<td>SMI - Array - Controllers</td>
<td>Retrieves controller information.</td>
</tr>
<tr>
<td>SMI - Array - Disks</td>
<td>Retrieves storage disk information.</td>
</tr>
<tr>
<td>SMI - Array - File Shares</td>
<td>Enumerates NAS file shares from a storage server.</td>
</tr>
<tr>
<td>SMI - Array - Pools</td>
<td>Retrieves storage pools.</td>
</tr>
<tr>
<td>SMI - Array - Ports</td>
<td>Retrieves storage ports.</td>
</tr>
<tr>
<td>SMI - Array - Volumes</td>
<td>Retrieves storage volumes.</td>
</tr>
<tr>
<td>SMI - Fabric</td>
<td>Retrieves SANs, fabrics, zone sets, zones, zone aliases, endpoints, and connections.</td>
</tr>
<tr>
<td>SMI - Fibre Channel Switch</td>
<td>Retrieves FC switches.</td>
</tr>
<tr>
<td>SMI - NAS Head - Component Systems</td>
<td>Retrieves all virtual file servers in a NAS head profile.</td>
</tr>
<tr>
<td>SMI - NAS Head - File Server IPs</td>
<td>Retrieves IP addresses for each NAS file server.</td>
</tr>
<tr>
<td>SMI - NAS Head - File Servers</td>
<td>Retrieves NAS file servers such as Common Internet File System (CIFS) and Network File System (NFS).</td>
</tr>
<tr>
<td>SMI - NAS Head - File Shares</td>
<td>Retrieves file shares for each NAS file server.</td>
</tr>
<tr>
<td>SMI - Storage Server</td>
<td>Retrieves SAN and NAS arrays and servers.</td>
</tr>
<tr>
<td>SMI - WBEM Service</td>
<td>Retrieves WBEM Service information such as profiles and SMI-S version.</td>
</tr>
</tbody>
</table>

Configure Discovery for a standalone storage device

Use this procedure for configuring a standalone storage device with the required SMI-S Provider for Discovery.

Role required: admin

1. On the storage device, if the SMI-S Provider is not present, install the SMI-S Provider software. The SMI-S Provider software is often part of the device management software. For more information, download the SMI-S Provider instructions from the storage provider manufacture. For example, if the SAN contains an EMC storage device, click [here](#) for more information.

2. For NetApp storage devices, install the SMI-S agent on the storage device host.

   **Note:** Discovery can also perform native discovery of NetApp servers without accessing the SMI-S server. See [NetApp server discovery](#) for more information.

3. Start the SMI-S Provider service.

4. In the SMI-S Provider or agent, configure the Discovery Interval with a synchronization rate that allows the wbem probe to receives the most current information during discovery. For example, for EMC storage, set the Discovery Interval.
EMC provider example

5. On the ServiceNow instance, set up **CIM credentials**.
6. Create a Discovery Schedule with the IP addresses of each SMI-S Provider.
7. Optionally, create a **Discovery behavior** that uses a functionality definition with a **wbem** port probe to make the initial port-scanning phase (Shazzam) more efficient.
8. Run a basic IP address Discovery.

Data collected by Discovery on storage devices
Discovery identifies and classifies information about storage devices.

<table>
<thead>
<tr>
<th>Label</th>
<th>Table name</th>
<th>Field name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sys ID</td>
<td>Storage Disk (cmdb_ci_storage_disk)</td>
<td>sys_id</td>
<td>N/A</td>
</tr>
<tr>
<td>File Share ID</td>
<td>Storage File Share (cmdb_ci_storage_fileshare)</td>
<td>fileshare_id</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Path</td>
<td>Storage File Share (cmdb_ci_storage_fileshare)</td>
<td>path</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Disk space (GB)</td>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
<td>disk_space</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Pool ID</td>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
<td>pool_id</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Label</td>
<td>Table name</td>
<td>Field name</td>
<td>Source</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------</td>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td>Speed (GFC)</td>
<td>Storage Port (cmdb_ci_storage_port)</td>
<td>speed</td>
<td>CIM probe</td>
</tr>
<tr>
<td>WWPN</td>
<td>Storage Port (cmdb_ci_storage_port)</td>
<td>wwpn</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Device ID</td>
<td>Storage HBA (cmdb_ci_storage_hba)</td>
<td>device_id</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Firmware version</td>
<td>Storage Server (cmdb_ci_storage_server)</td>
<td>firmware_version</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Disk Space (GB)</td>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>disk_space</td>
<td>CIM probe</td>
</tr>
<tr>
<td>LUN</td>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>lun</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Size (vdisk)</td>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>size</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Name (vdisk)</td>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>name</td>
<td>CIM probe</td>
</tr>
<tr>
<td>Object ID (vdisk)</td>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>object_id</td>
<td>CIM probe</td>
</tr>
</tbody>
</table>

### Storage relationships

Discovery establishes the correct relationships between Network-Attached Storage (NAS) storage devices and remotely mounted client servers that consume the storage. Discovery maps NAS file shares. It maps by taking the IP or hostname of a remote mounted disk on the client computer. It then matches it to the IP or hostname of the storage server providing the exported file system.

Discovery creates the following relationships for storage CIs running on Storage Area Networks.

#### SAN CI relationships

<table>
<thead>
<tr>
<th>Parent Component</th>
<th>Relationship</th>
<th>Child Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>Exports to::Imports from</td>
<td>Fibre Channel Disk (cmdb_ci_fc_disk)</td>
</tr>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>Exports to::Imports from</td>
<td>iSCSI (cmdb_ci_iscsi_disk)</td>
</tr>
</tbody>
</table>

Discovery maps NAS file shares. It maps by resolving the hostname of a remote mounted disk on the client computer to an IP address. It then matches it to the IP address of the storage server that provides the exported file system. Discovery extracts the hostname or IP address from the NAS path to determine the identity of the storage server. If the hostname is an actual hostname, the system immediately resolves that value into an IP address. It also stores it in the `nas_ip_address` field of the NAS File System (cmdb_ci_nas_file_system) table.

Discovery creates the following relationships for storage CIs running on Network Attached Storage (NAS). These relationships are the same between Linux and Windows operating system hosts.
NAS CI relationships

<table>
<thead>
<tr>
<th>Parent Component</th>
<th>Relationship</th>
<th>Child Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAS File System</td>
<td>Allocated from::Allocated to</td>
<td>Storage File Share</td>
</tr>
<tr>
<td>(cmdb_ci_nas_file_system)</td>
<td></td>
<td>(cmdb_ci_storage_fileshare)</td>
</tr>
</tbody>
</table>

NetApp server discovery

Discovery can find NetApp servers, using either CIM probes or patterns. The use of patterns, allows you to perform native discovery of the NetApp server without the need to traverse an SMI-S server. We recommend that the target NetApp server disable SSLv3 and enable TLS, since Java Runtime used by the MID Server does not allow SSLv3 by default. The key to encrypt the certificate must be at least 1024 bits.

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article [KB0694477](#) for more information.

NetApp server discovery options

<table>
<thead>
<tr>
<th>Method</th>
<th>Supported modes</th>
<th>Upgrades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery with patterns using ONTAP API</td>
<td>• NetApp 7-mode</td>
<td>Active by default for new instances.</td>
</tr>
<tr>
<td>The patterns support native discovery of NetApp servers, without requiring the need to discover the SMI-S server.</td>
<td>• NetApp cluster mode</td>
<td></td>
</tr>
<tr>
<td>Discovery with probes and sensors using CIM.</td>
<td>• NetApp 7-mode</td>
<td>Active by default for upgrades. To use patterns on upgraded instances, enable the relevant NetApp property (see below).</td>
</tr>
<tr>
<td></td>
<td>• NetApp cluster mode</td>
<td></td>
</tr>
</tbody>
</table>

NetApp discovery does not support the following:

- IP address and port redundancy
- Server Virtual Machines (SVM) redundancy within clusters
- Vserver tunnelling
- IPSpace
- The provisioning location

ONTAP version support

Discovery can find NetApp servers running these versions of the ONTAP operating system:

- For 7-mode: versions 8.3 and below.
- For cluster mode: version 8 or above.
NetApp discovery properties

To use native discovery with patterns, navigate to **Discovery Definition > Properties** and enable these properties:

<table>
<thead>
<tr>
<th>Property label</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>For 7-Mode NetApp storage servers use native discovery instead of SMI-S.</td>
<td>glide.discovery.sensors.netapp_native_7mode</td>
</tr>
<tr>
<td>For Cluster Mode NetApp storage servers use native discovery instead of SMI-S.</td>
<td>glide.discovery.sensors.netapp_native_cluster_mode</td>
</tr>
</tbody>
</table>

**Note:** If you already performed NetApp discovery, and then upgrade to madrid and enable these properties, a migration script runs the next time you perform NetApp discovery. This script could delete CIs for items like storage pools, which probes discover but the patterns do not. It also creates new relationships.

Credentials

<table>
<thead>
<tr>
<th>Credential type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure <strong>SNMP community credentials</strong> to classify the NetApp server.</td>
</tr>
<tr>
<td>Using local credentials, create a credential of type <strong>basic authentication</strong> to use when identifying and exploring the NetApp server. On the NetApp server, the credential must use the authentication method of &quot;password&quot; and allow read-only access to the ONTAP API. For cluster mode, the user can have a read-only role with default access. For 7-mode, the user must have a custom role with the following capabilities:</td>
</tr>
<tr>
<td>login_snmp, login-ndmp, login-sp, and login-http-admin</td>
</tr>
<tr>
<td>Api-*</td>
</tr>
</tbody>
</table>

Classifiers and patterns

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Probe</th>
<th>Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetApp Storage Server 7-Mode</td>
<td>Horizontal discovery probe: launches patterns</td>
<td>NetApp Storage 7-Mode</td>
</tr>
<tr>
<td>NetApp Storage Server Cluster-Mode</td>
<td></td>
<td>NetApp Storage Cluster-Mode</td>
</tr>
</tbody>
</table>

Running Discovery

To run discovery using these patterns, create a Discovery schedule that targets the management IP address of the NetApp server. Make sure that the basic authentication credentials that you configured can access the NetApp server.
Data collected by NetApp patterns

The following tables list the data that Discovery can find when you run the NetApp discovery patterns.

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_storage_server</td>
<td>name, firmware_version, cluster, serial_number, manufacturermode, model_id, cim_object_path, short_description, ip_address</td>
</tr>
<tr>
<td>cmdb_ci_storage_cluster</td>
<td>short_description, ip_address, manufacturer</td>
</tr>
<tr>
<td>cmdb_ci_storage_cluster_node</td>
<td>name, operational_status</td>
</tr>
<tr>
<td>cmdb_ci_storage_node_element</td>
<td>cluster, short_description, manufacturer, serial_number, model_id, ram, cpu_type, cpu_manufacturer, cpu_core_count</td>
</tr>
<tr>
<td>cmdb_ci_network_adapter</td>
<td>name, ip_address, netmask, cmdb_ci</td>
</tr>
<tr>
<td>cmdb_serial_number</td>
<td>serial_number, serial_number_type</td>
</tr>
<tr>
<td>cmdb_ci_ip_address</td>
<td>ip_address, netmask, nic, ip_version (Storage-7 mode), name</td>
</tr>
<tr>
<td>Table</td>
<td>Fields</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>cmdb_ci_disk</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>device_id</td>
</tr>
<tr>
<td></td>
<td>serial_number</td>
</tr>
<tr>
<td></td>
<td>manufacturer</td>
</tr>
<tr>
<td></td>
<td>model_id</td>
</tr>
<tr>
<td></td>
<td>storage_type</td>
</tr>
<tr>
<td></td>
<td>device_interface</td>
</tr>
<tr>
<td></td>
<td>size_bytes</td>
</tr>
<tr>
<td></td>
<td>computer</td>
</tr>
<tr>
<td>cmdb_ci_storage_fileshare</td>
<td>path</td>
</tr>
<tr>
<td></td>
<td>name</td>
</tr>
<tr>
<td>cmdb_ci_storage_pool</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>short_description</td>
</tr>
<tr>
<td></td>
<td>free_space_bytes</td>
</tr>
<tr>
<td></td>
<td>pool_id</td>
</tr>
<tr>
<td></td>
<td>size_bytes</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>lun</td>
</tr>
<tr>
<td></td>
<td>volume_id</td>
</tr>
<tr>
<td>cmdb_ci_iscsi_export</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>lun</td>
</tr>
<tr>
<td></td>
<td>initiator_iqn</td>
</tr>
<tr>
<td></td>
<td>iqn</td>
</tr>
<tr>
<td></td>
<td>export_id</td>
</tr>
<tr>
<td>cmdb_ci_storage_controller</td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>device_id</td>
</tr>
<tr>
<td>cmdb_ci_fc_export</td>
<td>initiator_wwpn</td>
</tr>
<tr>
<td></td>
<td>export_id</td>
</tr>
<tr>
<td></td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>lun</td>
</tr>
<tr>
<td>cmdb_ci_fc_port</td>
<td>speed</td>
</tr>
<tr>
<td></td>
<td>name</td>
</tr>
<tr>
<td></td>
<td>wwnn</td>
</tr>
<tr>
<td></td>
<td>wwpn</td>
</tr>
</tbody>
</table>
Manage large storage payloads in Linux and Solaris

Large payloads for Linux and Solaris direct attached storage can cause out of memory errors if not configured to serialize the processing of the payload.

Role required: admin or discovery_admin

When Linux or Solaris storage sensors process payloads from large disk arrays, worker threads run simultaneously and can run an instance node out of memory, requiring a restart. To prevent this, the \( \text{discovery_sensor_yield} \) table controls the behavior of worker threads used to process large payloads. When Discovery processes a sensor, it checks the \( \text{discovery_sensor_yield} \) table for that sensor. If the sensor is listed in the table, and if the payload size exceeds the configured limit in the table, then the worker attempts to secure a lock before processing the payload. If the lock is not available because another worker is already processing a large payload, the waiting worker reschedules the next job for 30 seconds later and moves on to other tasks. The job then waits in the Schedule Item (sys_trigger) table for an idle worker to pick it up. This serialization occurs per node, enabling multiple nodes to process large payloads simultaneously. The processing interval of 30 seconds is not configurable.

Caution: These threshold values were created for Linux and Solaris storage sensors only and, in most cases, should not be changed. If your instance is running out of memory while simultaneously processing multiple, large storage payloads, consider scheduling Discovery of these systems to times of lower activity. If this is not possible, examine the ECC queue to determine which storage sensor might be causing the issue and note the size of the payload. If the payload size is below that of the threshold in the \( \text{discovery_sensor_yield} \) table, lower the threshold value enough to serialize that payload. If a sensor other than the Linux or Solaris storage sensor is responsible, you can add that sensor to the \( \text{discovery_sensor_yield} \) table. Make sure to set the threshold value correctly for the payload size. Serializing sensor processing can slow Discovery.

1. In the application navigator filter, enter \text{discovery_sensor_yield.list}.
   The base Discovery system serializes these sensors when their payloads exceed 512 KB:
   - Solaris - Storage
   - Linux - Storage

2. To change the threshold, double-click in the \text{Value} column in the list and edit the size.
Editing the sensor yield threshold

3. To add another sensor and configure it for payload serialization, click **New** and complete the fields in the form.
   - **Sensor**: Storage sensor to configure.
   - **Value**: Threshold size for the sensor.

4. Click **Submit**.

**Storage discovery examples**
Discovery creates configuration items (CI) and CI relationships for physical and logical storage components attached directly to application and database servers or by fibre channel switched fabric in a multi-path configuration.

**Direct attached storage**

In this example of direct attached storage (DAS), a SCSI drive with two partitions, `/dev/sda1` and `/dev/sda2`, is attached to a Linux host. The `/dev/sda1` partition is bootable and supports the system software. The `/dev/sda2` partition contains a logical volume configured as a storage pool and mounted to the Linux root file system by Logical Volume Management (LVM). The storage pool uses only 1.5GB of the partition, leaving 13GB of storage available for additional logical volumes.
### Direct attached storage (DAS) example

**CIs and relationships created for direct attached storage (DAS)**

<table>
<thead>
<tr>
<th>Configuration item</th>
<th>Description</th>
<th>Tables</th>
<th>Key reference and Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda</td>
<td>SCSI physical storage device</td>
<td>(cmdb_ci_disk), (cmdb_ci_storage_device), (cmdb_rel_ci)</td>
<td>Provides: /dev/sda, Provided by: /dev/sda1</td>
</tr>
</tbody>
</table>

- Provides: /dev/sda
- Provided by: /dev/sda1
<table>
<thead>
<tr>
<th>Configuration item</th>
<th>Description</th>
<th>Tables</th>
<th>Key reference and Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda1</td>
<td>Partition 1 on the SCSI storage device</td>
<td>{cmdb_cl_partition, cmdb_cl_storage_volume, cmdb_cl_file_system, cmdb_rel_cl}</td>
<td>{cmdb_cl_file_system}</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Mount point: /boot</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>File system: Ext4</td>
</tr>
<tr>
<td>/dev/sda2</td>
<td>Partition 2 on the SCSI storage device</td>
<td>{cmdb_cl_partition, cmdb_cl_storage_volume, cmdb_cl_file_system, cmdb_rel_cl}</td>
<td>{cmdb_cl_storage_pool_member}</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Pool: /dev/mapper/lvm-root-333-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Storage: /dev/sda2</td>
</tr>
<tr>
<td>/dev/mapper/lvm-root-333-0</td>
<td>Linux logical volume, mapped with LVM to a physical disk storage partition.</td>
<td>{cmdb_cl_storage_device, cmdb_cl_storage_volume, cmdb_cl_file_system, cmdb_cl_storage_pool, cmdb_cl_lvm_pool, cmdb_cl_lvm_pool_member, cmdb_rel_cl}</td>
<td>{cmdb_cl_storage_pool_member}</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Pool: /dev/mapper/lvm-root-333-0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Storage: /dev/sda2</td>
</tr>
</tbody>
</table>

**Multipath fibre channel storage**

In this example of a fibre channel storage area network (SAN), two physical storage devices, mpatha and mpathb, are attached to a Linux host through fibre switches, which provide failover capabilities. The mpatha drive contains two partitions, mpatha1 and mpatha2. The first partition is mounted directly to /boot on the Linux host. Three logical volumes are mapped to the mpatha2 partition and to the physical device mpathb. The logical volumes are mounted as Ext4 file systems in folders on the Linux root structure. This example shows the CIs that Discovery manages for each component and the mounting points for the logical volumes on the Linux host.
Switched fibre fabric details

Discovery creates CIs for the logical sub-components in NAS and SAN environments, such as fibre channel disks and pool components, as well as for host bus adapters (HBA) and physical block storage. In multipath environments, Discovery creates CI relationships within the switched fibre fabrics that connects the Linux host to the physical storage devices. In this diagram, the fibre fabrics have redundant paths that the SAN environment can use for failover if connections fail.
Operating-system-level virtualization (OSLV) discovery

Discovery can collect image and container information from OSLV engines. Discovery can determine container status and size and identify images and their tags provided by operating-system-level virtualization.

Note: Currently, the ServiceNow® platform supports the discovery of Docker virtualization containers only.

Table schema

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_oslv_engine</td>
<td>Base table for all OSLV engines.</td>
</tr>
<tr>
<td>cmdb_ci_oslv_image</td>
<td>Base table for globally unique OSLV images.</td>
</tr>
<tr>
<td>cmdb_ci_oslv_local_image</td>
<td>Base table for local instances of OSLV images.</td>
</tr>
<tr>
<td>cmdb_ci_oslv_image_tag</td>
<td>Base table for tags on OSLV images.</td>
</tr>
<tr>
<td>cmdb_ci_oslv_container</td>
<td>Base table to store containers found on the host.</td>
</tr>
</tbody>
</table>

Data collected

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System Level Virtualization Engine (cmdb_ci_oslv_engine)</td>
<td>Server (server)</td>
</tr>
<tr>
<td>Operating System Level Virtualization Image (cmdb_ci_oslv_image)</td>
<td>• Image ID (image_id)</td>
</tr>
<tr>
<td></td>
<td>• Image digest (image_digest)</td>
</tr>
<tr>
<td></td>
<td>• Image created (image_created_at)</td>
</tr>
<tr>
<td></td>
<td>• Size (bytes) (size_bytes)</td>
</tr>
<tr>
<td>Operating System Level Virtualization Local Image (cmdb_ci_oslv_local_image)</td>
<td>Image ID (image_id)</td>
</tr>
<tr>
<td>Operating System Level Virtualization Image Tag (cmdb_ci_oslv_image_tag)</td>
<td>• Image ID (image_id)</td>
</tr>
<tr>
<td></td>
<td>• Repository (repository)</td>
</tr>
<tr>
<td></td>
<td>• Tag (tag)</td>
</tr>
<tr>
<td>Operating System Level Virtualization Container (cmdb_ci_oslv_container)</td>
<td>• Container ID (container_id)</td>
</tr>
<tr>
<td></td>
<td>• Status (status)</td>
</tr>
<tr>
<td></td>
<td>• Size (bytes) (size_bytes)</td>
</tr>
<tr>
<td></td>
<td>• Image ID (image_id)</td>
</tr>
</tbody>
</table>

Docker virtualization

Discovery can collect data about specific objects in a Docker engine, running on a Linux host. The ServiceNow® platform supports the discovery of Docker release 1.11.0 or later.
Docker configuration items

Discovery creates configuration items (CI) for these Docker objects:

- **Engine**: Software that runs on the Linux host to create the operating environment for distributed applications.
- **Images**: Images on the Docker engine separated into these entities:
  - Global images.
  - Locally stored instances of the images.
  - Tags associated with the locally stored instances of the images.
- **Containers**: Virtual wrappers, found on the Docker engine, which contain running instances of images.

Docker restrictions and considerations

When using Docker virtualization, consider the following:

- The initial Discovery process scan can identify an application in a container and classify it correctly. However, subsequent probes launched to explore that application cannot see inside the container and cannot return details about the application.
- Discovery scans all containers it finds, including inactive containers, which can slow down Discovery. You should delete containers that are not running.
- Only one Docker engine is permitted per computer (on either a physical or virtual machine).

User privileges

The user whose credentials are used to perform Docker Discovery must have privileges defined by one of these methods:

- Provide a user with elevated rights for running commands, since the Docker daemon runs as the root user. The Docker pattern supports the use of privileged commands, such as `sudo` or `pbrun`, to run as the root user.
- Assigned to a group named `docker`, which has special privileges for running Docker commands. For instructions on setting up a group, see [Create a Docker group](#).

Table schema and relationships

The Docker tables are installed with the Discovery plugin and extend tables used to store data on the operating-system-level virtualization (OSLV) engine.

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docker Engine (cmdb_ci_docker_engine)</td>
<td>Stores instances of the Docker engine.</td>
</tr>
<tr>
<td>Docker Image (cmdb_ci_docker_image)</td>
<td>Stores the globally unique representation of Docker images.</td>
</tr>
<tr>
<td>Docker Local Image (cmdb_ci_docker_local_image)</td>
<td>Stores local instances of Docker images.</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>Docker Image Tag (cmdb_ci_docker_image_tag)</td>
<td>Stores tags from local Docker images.</td>
</tr>
<tr>
<td>Docker Container (cmdb_ci_docker_container)</td>
<td>Stores Docker containers found on the host.</td>
</tr>
</tbody>
</table>

Discovery stores these relationships in the CI Relationship (cmdb_rel_ci) table.

- `cmdb_ci_server` Runs::RunUnder `cmdb_ci_docker_engine`
- `cmdb_ci_docker_engine` Manages::ManagedBy `cmdb_ci_docker_container`
- `cmdb_ci_docker_engine` Manages::ManagedBy `cmdb_ci_docker_local_image`
- `cmdb_ci_docker_container` Instantiates::InstantiatedBy `cmdb_ci_docker_local_image`
- `cmdb_ci_docker_image_tag` RegisteredOn::HasRegistered `cmdb_ci_docker_local_image`
- `cmdb_ci_docker_local_image` Instantiates::InstantiatedBy `cmdb_ci_docker_image`
Parent and child relationships for the OSLV and Docker dependent tables
Identification, containment, and hosting rules

Discovery uses an application rule identifiers to find the Docker engine and then applies other rules to identify specific Docker objects.

Application rule identifier

The system creates the cmdb_ci_docker_engine configuration item (CI) during process classification. Based on this, Discovery uses the Application Rule identifier, on the Application [cmdb_ci_appl] table to identify the particular Docker engine encountered. After establishing this identity, Discovery uses the relationships defined in the containment and hosting rules to accurately create and update the individual Docker component CIs related to that engine.

Identifiers

<table>
<thead>
<tr>
<th>Name</th>
<th>Table</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docker Container</td>
<td>Docker Container</td>
<td>container_id</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_docker_container)</td>
<td></td>
</tr>
<tr>
<td>Docker Global Image</td>
<td>Docker Image</td>
<td>image_id</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_docker_image)</td>
<td></td>
</tr>
<tr>
<td>Docker Local Image</td>
<td>Docker Local Image</td>
<td>image_id</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_docker_local_image)</td>
<td></td>
</tr>
<tr>
<td>Docker Image Tag</td>
<td>Docker Image Tag</td>
<td>repository, tag</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_docker_image_tag)</td>
<td></td>
</tr>
</tbody>
</table>

Containment and hosting rules

Docker Discovery uses these containment and hosting rules to create configuration items (CI) from the data returned by the Docker Pattern. After Discovery identifies the Docker engine by its relationship to the Application (cmdb_ci_appl) table, it uses these rules to identify the specific CIs connected to that engine from their relationships to one another. By connecting the components to one another in this fashion, from the application down, starting with the engine, Discovery avoids creating duplicate CIs for components from other Docker engines that use the same name or image_id.

Containment rule

<table>
<thead>
<tr>
<th>Parent</th>
<th>Child</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docker Local Image</td>
<td>Docker Image Tag</td>
<td>Has registered</td>
</tr>
</tbody>
</table>

Hosting rules

<table>
<thead>
<tr>
<th>Parent</th>
<th>Child</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docker Container</td>
<td>Docker Engine</td>
<td>Managed by</td>
</tr>
<tr>
<td>Docker Local Image</td>
<td>Docker Engine</td>
<td>Managed by</td>
</tr>
</tbody>
</table>
Pattern

To view the pattern for discovering the docker engine (cmdb_ci_docker_engine) and its components, navigate to Pattern Designer > Discovery Patterns and open Docker Pattern. For more information about Discovery patterns, see Discovery patterns used by ITOM Visibility.

Discovery runs the Docker Engine process classifier in the network. If the classifier identifies the dockerd or docker daemon process, the classifier triggers the Horizontal Pattern (HorizontalDiscoveryProbe) probe, which launches the Docker Pattern and begins collecting data from Docker components.

The pattern collects data for the main CI type, cmdb_ci_docker_engine, and for these related CI types:

- cmdb_ci_docker_image
- cmdb_ci_docker_local_image
- cmdb_ci_docker_image_tag
- cmdb_ci_docker_container

**Note:** Probe-to-pattern migration is supported for the New York release and later. See the knowledge article KB0694477 for more information.

Data collected

These attributes are discovered, in addition to the attributes derived from the parent OSLV tables.

<table>
<thead>
<tr>
<th>Table</th>
<th>Fields</th>
</tr>
</thead>
</table>
| Docker Engine (cmdb_ci_docker_engine) | - Version (version)  
- Go version (go_version)  
- Git commit (git_commit)  
- Build date (build_date)  
- OS/Arch (os_arch)  
- API version (api_version) |
| Docker Image (cmdb_ci_docker_image) | No additional attributes are discovered for this child table. |
| Docker Local Image (cmdb_ci_docker_local_image) | No additional attributes are discovered for this child table. |
| Docker Image Tag (cmdb_ci_docker_image_tag) | No additional attributes are discovered for this child table. |
| Docker Container (cmdb_ci_docker_container) | - Image ID (image_id)  
- Command (command)  
- Container created (container_created_at) |

Discovery resource utilization

Typical transactions on Windows and Unix generate various amounts of network traffic, depending on what is being discovered.
This table shows the bandwidth consumption, by operating system, for each data flow segment of a typical Discovery using probes and patterns; the devices use probes while the applications use patterns. Also shown is the bandwidth comparison between an initial Discovery for three-tier applications and each subsequent Discovery that does not collect hardware data. All measurements are in megabytes.

Note: These measurements were taken with base operating configurations. Your local system results may vary.
### OS Discovery Using Patterns

<table>
<thead>
<tr>
<th>Device Type</th>
<th>MID &gt; Instance</th>
<th>Instance &gt; MID</th>
<th>Mid &gt; Target</th>
<th>Target &gt; MID</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 2016</td>
<td>0.104966</td>
<td>0.101271</td>
<td>0.77739</td>
<td>2.364353</td>
<td>3.34798</td>
</tr>
<tr>
<td>Windows 2012</td>
<td>0.126327</td>
<td>0.07928</td>
<td>1.177146</td>
<td>3.707051</td>
<td>5.089804</td>
</tr>
<tr>
<td>Windows 2008</td>
<td>0.141816</td>
<td>0.104674</td>
<td>1.032673</td>
<td>3.594784</td>
<td>4.873997</td>
</tr>
<tr>
<td>Windows 10</td>
<td>0.091466</td>
<td>0.075601</td>
<td>0.642313</td>
<td>2.221103</td>
<td>3.030483</td>
</tr>
<tr>
<td>Linux CentOS</td>
<td>0.164232</td>
<td>0.111376</td>
<td>0.148742</td>
<td>0.690117</td>
<td>1.114467</td>
</tr>
<tr>
<td>Mac OS X</td>
<td>0.103707</td>
<td>0.068332</td>
<td>0.021681</td>
<td>0.461365</td>
<td>0.655055</td>
</tr>
<tr>
<td>HP-UX</td>
<td>0.120358</td>
<td>0.106676</td>
<td>0.042665</td>
<td>0.101149</td>
<td>0.370852</td>
</tr>
<tr>
<td>Solaris</td>
<td>0.130551</td>
<td>0.099414</td>
<td>0.060243</td>
<td>0.346605</td>
<td>0.636813</td>
</tr>
<tr>
<td>Cisco UCS switch</td>
<td>0.029665</td>
<td>0.027465</td>
<td>0.094918</td>
<td>0.097444</td>
<td>0.240492</td>
</tr>
<tr>
<td>FS load balancer</td>
<td>0.043935</td>
<td>0.03689</td>
<td>0.017179</td>
<td>0.012132</td>
<td>0.110136</td>
</tr>
<tr>
<td>A10 load balancer</td>
<td>0.046631</td>
<td>0.032266</td>
<td>0.018311</td>
<td>0.01182</td>
<td>0.12903</td>
</tr>
<tr>
<td>EMC Storage</td>
<td>0.4776</td>
<td>0.373828</td>
<td>1.215954</td>
<td>4.741926</td>
<td>6.809308</td>
</tr>
</tbody>
</table>

**Three-tier Application - Initial Discovery**
- FS load balancer: 0.712829, 0.678862, 7.084678, 9.430181, 17.90655
- Apache on Linux: 0.021779, 0.012132
- WebSphere on Linux: 0.540161, 1.107108
- Oracle on Windows: 0.729403, 1.165112
- Total: 5.797945, 7.145829

**Three-tier Application - Subsequent Discovery**
- FS load balancer: 0.150882, 0.107409, 2.536535, 0.560122, 3.354948
- Apache on Linux: 0.003347, 0.012132
- WebSphere on Linux: 0.136366, 0.079392
- Oracle on Windows: 0.341042, 0.11365
- Total: 2.05778, 0.354948

This table shows discovery of different OS types using patterns.
Discovery monitoring and issue resolution

Learn how to monitor the progress of your discoveries and how to configure the system to aggregate performance metrics that are important to you. Find descriptions of the error messages you see, as well as possible steps you can take to solve problems. The Knowledge Base on Hi contains several articles to help you troubleshoot discovery issues.

*Discovery Home Page*

Open the Discovery Manager to create or edit schedules for IP-based and Cloud Discovery. View the CI's and cloud resources that your schedules discover and then track any errors that occurred.

*Discovery Dashboard*

Monitor ongoing Discovery operations and manage reports that query the database. View summary information about Discovery status, discovered devices, timelines, and errors.

*Discovery Troubleshooting Guide*

Look at the Discovery troubleshooting guide in the knowledge base if you are experiencing symptoms, such as the inability for Discovery to classify a discovered host, or the Shazzam probe not finding certain types of devices.

*Discovery error messages*

View the list of common error messages and possible solutions.

*Discovery performance metrics*

View performance metrics on probe/pattern and sensor processing times. You can use the roll-up data to monitor the performance of specific discoveries or to compare performance between versions after an upgrade.

**Discovery Home page**

The Discovery Home page provides a summary of discoveries that were triggered by cloud and non-cloud schedules. You can view any errors that occurred during a Discovery and find
remediation suggestions. Use the Home page view to examine the cloud resources discovered for the service accounts you selected in a cloud schedule.

To open the Discovery Home page, navigate to Discovery > Home. The Home page shows the Discovery totals for each of the following categories. Click a tile to see details for that category.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedules</td>
<td>The number of Discovery schedules that are configured to find cloud resources and virtual machines (IP addresses). Click View Schedules to open the CI schedule view, where you can see specific details about each CI schedule separately. View the results of your cloud or IP address schedules and create new schedules.</td>
</tr>
<tr>
<td>Discovered Devices</td>
<td>The total number of device CIs discovered. Devices include all the CIs in tables that extend the Hardware (cmdb_ci_hardware) table, such as Computer (cmdb_ci_computer) and Cisco UCS Equipment (cmdb_ci_ucs_equipment).</td>
</tr>
<tr>
<td>Cloud Resources</td>
<td>The total number of resources (virtual machines) discovered for the service accounts selected. See Enable virtual machine Discovery in your cloud environment for details about setting up Discovery for your cloud resources. Click View Cloud Resources to open a list of discovered cloud resources, by type.</td>
</tr>
<tr>
<td>Errors</td>
<td>The total number of unresolved errors that occurred during discovery. Click View Errors to drill down to specific types of errors.</td>
</tr>
</tbody>
</table>

**Viewing all schedules**

The schedule view displays details for each schedule that ran for the selected filter. By default, the list shows all the Discovery schedules in alphabetical order, regardless of their current running status.
Use these controls to filter Discovery results by schedules:

**Schedule filters**

You have the choice of these filters:
- **All**: Results of all Discovery schedules that ran. This is the default filter.
- **Cloud Schedules**: Results of all Cloud Discovery schedules that ran. When you select this filter, a second filter appears, allowing you to display cloud schedules by **Service Account**.

![Cloud Schedules](image)

- **IP Based Schedules**: Results of all Discovery schedules that ran for Configuration Items (CI), using IP ranges. When you select this filter, a second filter appears, allowing you to display the CI schedules by a specific location or all locations.

![IP Based Schedules](image)

**Schedule sorter**

From the schedule selector in the left pane, select a sorting criteria, from left to right:

- Errors for each schedule listed.
- Unidentified IP addresses.
- Recent activity, which lists the schedules that are currently running first.
- Alphabetical listing of schedules.

**Schedule list**

List of available Discovery schedules. Select a schedule to display its details. Click the pencil icon at the top of the page to edit the schedule.
**Date/time indicator**

Date and time the selected schedule ran.

**Schedule topics**

Tiles that show the Discovery result categories for the selected schedule. Click a tile to see more detail.

- **Cloud resources**: Displays the total number of CIs updated or created for this filter.
- **Virtual Machines**: Displays the total number of virtual machines discovered with the selected schedule.
- **Errors**: Displays the total number of errors that occurred from running the selected schedule.
- **Datacenters**: Displays the number of datacenters selected during the creation of the Discovery schedule.

**Cloud resources trend chart**

Trend graph for the selected schedule. Each circle in the graph represents a specific Discovery run.

Place the cursor over any circle to see a summary of that Discovery.

Click the circle to filter the values you see in the Discovery topics. The results populate the appropriate tiles above the graph.

**Errors by category**

Category tiles for the errors encountered during the Discovery for the selected schedule. Click a tile to view the errors in that category and to see recommended actions for remediation. See *Discovery error messages* for additional details.

**Discovered cloud resources**

Breakdown of the cloud resources discovered for the selected schedule, presented in a horizontal bar chart.

**Add schedules**

Types of schedules to add.

**Advanced View**
Displays the Discovery status record for the selected schedule. Use this link to examine the results of the schedule in the ECC Queue and the Discovery log.

**Viewing IP-based schedules**

To display results from IP range discoveries, select **IP Based Schedules** in the primary filter. Use the secondary field that appears to filter the search results by **Location**. The tiles show the result categories for the selected IP-based schedule. Click a tile to see more detail.

- **Discovered devices**: Total count of virtual machines discovered by the IP-based schedule, including both targeted and unidentified IP addresses.
- **Errors**: Displays the total number of errors encountered running the selected IP-based schedule.
- **Unidentified IPs**: Count of unidentified CIs discovered by the IP-based scheduled.
- **Targeted IPs**: Count of CIs found that were configured in the IP-based schedule.
### Discovery Schedules

**CheckOne**  
**Unassigned Location**

#### Duration: 1m

<table>
<thead>
<tr>
<th>Discovered Devices</th>
<th>Errors</th>
<th>Unidentified IPs</th>
<th>Targeted IPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Discovered Devices Trend**

- Last 10 runs
  - 2019-05-30: 0
  - 2019-06-10: In progress
  - 2019-05-30: Completed

**Advanced View**
Resolve CI Discovery Schedule errors
From the ServiceNow Home page, you can view the Discovery errors that occurred during a Discovery and get suggestions for resolving these errors. You can view the errors for all schedules or for a single schedule.

Role required: discovery_admin

1. Navigate to Discovery > Home.
The summary page for all schedules appears.

2. View the errors, using one of these methods:
   - Click the Errors tile to see the errors by category for all CI discovery schedules.
   - Click the Schedules tile to display the errors for a single schedule.

1. In the summary screen, select the schedule that you want to investigate for errors.
   If you have numerous schedules, you can filter by location, using the location dropdown list, or filter by the schedule name in the Search by name field.

2. After you select the schedule, click the Errors tile.
3. Scroll down to view the errors by category.
3. Click a category tile to display the specific error codes that occurred in that category.

4. Click the tile for the error code you want to investigate.

The list displays all occurrences of that error code, either for a single CI schedule or for all schedules, depending on the path you took to view the errors. The list shows the IP address of each instance that experienced the error and the error status **Active error**. The **Recommended Actions** pane on the right contains the recommended action for the selected error. The **ACTION ON ALL** pane contains operations you can execute on all the errors in the list.
5. Click **View instructions** to see details and remediation suggestions for the error code.
6. To begin work, either select the check box next to an error to resolve that specific error or select an action for **ALL** to resolve that error code for all instances of that error. In this example, the system recommends creating new Windows credentials to resolve this error code.
7. Complete the suggested remediation tasks before attempting to retry Discovery. In this example, the **Create new Windows Credentials** link opens a form for creating and testing new Windows credentials.
8. After all the recommended tasks are complete, click **Retry Discovery**.
   A possible course of action might be to retry Discovery for **ALL** instances after creating the credentials. The new credentials might solve the issue for all the instances experiencing this error code.

9. Check the **Error Status** column to see the results of your remediation efforts.
   Single retries complete quickly. Retries for large numbers of IP addresses can take several minutes.

An error can have one of these statuses:

- **Active error**: Unresolved error. This is the default status of all new errors.
- **Resolved**: Resolved error. The recommended actions show **Error successfully resolved**.
- **Assigned**: Task assigned to resolve this error. Click **Create a task** to assign a user to a task for all active errors. Assigned errors are considered **Active errors**, and actions can still be performed on them.
- **Pending Discovery**: Waiting for Discovery to start after you execute the **Retry Discovery** action on all errors.
- **In Discovery**: Discovery is currently active.
Discovery Dashboard

The Discovery Dashboard is a central place for Discovery users to monitor ongoing Discovery operations. The dashboard contains reports that query the database and display the results.

Dashboards are the home pages for products on the instance. The Discovery Dashboard provides summary information about your Discovery status, devices, applications, and timeline. It also displays Discovery errors and suggested solutions.

**Note:** All the data on the dashboard is domain separated, and shows only data collected from the domain specified in the **MID Server user account** for the Discovery schedule used.

To display the Discovery Dashboard, navigate to **Discovery > Dashboard**.

**Features of the Discovery dashboard**

The Discovery Dashboard displays:
- The current progress (status) of the actively running Discovery schedules.
- Newly discovered devices and applications, by time discovered.
- Existing devices and applications not updated by Discovery.
- Errors that occurred during a Discovery schedule that has run.
- Credentials that were either not required or unused by a Discovery schedule.

**Adding Discovery reports to the dashboard**

Starting with the Madrid release, **gauges** are no longer supported in dashboards and have been replaced with **widgets**. You can continue using Discovery gauges on existing dashboards. However, you cannot add old Discovery gauges to your dashboards. If you need some reporting capability not offered in the current dashboard, you must create your own reports using widgets.

**Discovery Dashboard components**

You can customize the reports on the dashboard to display statistics important to your environment.

**Active Discovery Status report**

The Active Discovery Status report displays the progress of an actively running Discovery. The information displayed is pulled from the Discovery Status module. The **Progress** field of the status record specifies the percent complete of the Discovery schedule. The percent complete is calculated based on previous runs of the same Discovery schedule. The progress field is empty in the following cases:
- When you use DiscoverNow
- When you use Discovery from a configuration item (CI).
- During the first run of a Discovery schedule.
Active Discovery Status report

Newly Discovered Devices report

The Newly Discovered Devices report displays the count of the devices that the Discovery application has identified in the last 7 days, grouped by the class of the device.
Newly Discovered Devices report

Total Discovered Devices report

The Total Discovered Devices report displays the total count of the devices that the Discovery application has identified in the last 30 days, grouped by the class of the device.
Total Discovered Devices report

Unrefreshed Devices report

The Unrefreshed Devices report displays devices that were discovered up to a year ago, but have not been identified in the last 30 days, grouped by the class of the device.
Unrefreshed Devices report

Newly Discovered Applications report

The New Applications Discovered report displays the count of new applications that the Discovery application has identified in the last 7 days, grouped by the class of the application.
Newly Discovered Applications report

Total Discovered Applications report

The Total Discovered Applications report displays the total count of the applications that the Discovery application has identified in the last 30 days, grouped by the class of the application.
Total Discovered Applications report

Unrefreshed Applications report

The Unrefreshed Applications report displays applications that were discovered up to a year ago, but have not been identified in the last 30 days, grouped by the class of the application.
Unrefreshed Applications report

Active Discovery Errors report

The Active Discovery Errors report lists the errors that are generated by each active Discovery. The errors for a specific Discovery are overwritten when that Discovery runs again. This report displays the following data:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created</td>
<td>Timestamp of the error message.</td>
</tr>
<tr>
<td>Short message</td>
<td>Contents of the error message generated during the Discovery run.</td>
</tr>
<tr>
<td>Help</td>
<td>Link to a suggestion on how to fix the error. In some cases, the link will be specific to the error. In other cases, the link points to a general page that has the solution to a collection of common errors.</td>
</tr>
<tr>
<td>IP</td>
<td>IP address of the asset on which the error was detected.</td>
</tr>
</tbody>
</table>
## Column | Description
--- | ---
Discovery status | Discovery status number.

### Active Discovery Errors report

<table>
<thead>
<tr>
<th>Created</th>
<th>Short message</th>
<th>Help</th>
<th>IP</th>
<th>Discovery status</th>
</tr>
</thead>
<tbody>
<tr>
<td>2019-03-26 11:12:33</td>
<td>Error when processing Discovery: No sensors defined</td>
<td>Help link</td>
<td>10.0.0.1</td>
<td>Discovery status</td>
</tr>
<tr>
<td>2019-03-26 11:12:34</td>
<td>Error when processing Windows, Ubuntu, NullProbe script error for probe: ‘Windows · Network’ · Reference:Err: “related_data” is not defined.</td>
<td>Help link</td>
<td>10.0.0.1</td>
<td>Discovery status</td>
</tr>
</tbody>
</table>

### Unused Credentials report

The Unused Credentials report lists the credentials that are not being used by the Discovery or Orchestration applications at the current time.

**Note:** This may include credentials for a Discovery that has been scheduled but not run for the first time.
<table>
<thead>
<tr>
<th>Name</th>
<th>User name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>administrator</td>
<td>Windows</td>
</tr>
<tr>
<td>socrates</td>
<td>administrator</td>
<td>SSH</td>
</tr>
<tr>
<td>administrator</td>
<td>administrator</td>
<td>Windows</td>
</tr>
<tr>
<td>root</td>
<td>root</td>
<td>SSH</td>
</tr>
<tr>
<td>Valley</td>
<td>Valley</td>
<td>Windows</td>
</tr>
<tr>
<td>discolla1</td>
<td>discolla1/administrator</td>
<td>Windows</td>
</tr>
<tr>
<td>Cim</td>
<td>root</td>
<td>CIM</td>
</tr>
<tr>
<td>operations</td>
<td>operations</td>
<td>SSH</td>
</tr>
<tr>
<td>sftp</td>
<td>sftp/administrator</td>
<td>Windows</td>
</tr>
<tr>
<td>sftp_user</td>
<td>sftp_user</td>
<td>SSH</td>
</tr>
<tr>
<td>vcenter1</td>
<td>vcenter@vcenter1</td>
<td>VMWare</td>
</tr>
</tbody>
</table>

Actions on selected rows...
Discovery performance metrics

This Discovery enhancement collects performance metrics on probe/pattern and sensor processing times and then aggregates that data over time. You can use the roll-up data to monitor the performance of specific discoveries or to compare performance between versions after an upgrade.

Metrics

Discovery provides these individual performance metrics:

- Probe and pattern processing time.
- Sensor processing time.
- Identification and Reconciliation Engine (IRE) processing time for Discovery patterns. This processing time is already included in the sensor processing time, but is isolated here to provide more insight into the identification and reconciliation of pattern payloads.

Discovery can aggregate individual metrics for these attributes:

- Builds/versions
- Discovery status
- Target IP address

How metric aggregations are triggered

Metric roll-ups are initiated as follows:

- **Aggregated by build**: Implemented by the *Aggregate Discovery Probe And Sensor Metrics By Build* scheduled job. This job runs at 0200, local time.
- **Aggregated by status**: Implemented by the *Rollup Probe/Sensor Metrics by Status* Script Action, which is triggered by the `discovery.complete` or `discovery.cancelled` registered events.
- **Aggregated by target**: Implemented by the *Rollup Probe/Sensor Metrics by Target* Script Action that is triggered by the `discovery.device.complete` registered event.

**Note:** If Discovery execution is cancelled before it completes, Discovery cannot update the IP target metric aggregation table. This is because the `discovery.device.complete` event that triggers aggregation does not run. IP target data for an interrupted Discovery is collected when subsequent discoveries run successfully on the target. Cancelling Discovery execution does not affect the aggregation of other metrics, which are triggered differently.

Tables

Discovery performance metrics data is stored in these tables:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe and Sensor Metrics (Individual)</td>
<td>Stores the individual performance metrics for probes/patterns, sensors, and IRE processing times.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Discovery properties

Performance metrics properties control whether or not aggregation occurs, but not what data is included in the aggregation. Status and IP target data is collected as follows:

- Rollups for status always contain new data. Discovery continuously collects data on all probes and sensors during the discovery execution for that discovery status and stores it in the Probe and Sensor Metrics (Individual) (discovery_perf_metric_probe_sensor) table. Aggregation rolls up all probe and sensor data for that particular status after `discovery.cancel` and `discovery.complete` events are fired for that status, but only if the aggregation property for status roll-ups is enabled.

- Discovery continuously collects data on IP targets and stores it in the Probe and Sensor Metrics (Individual) (discovery_perf_metric_probe_sensor) table. Aggregation rolls up all existing IP target data after the `glide.discovery.perf.metrics.rollup_by_target` property is enabled and creates records in the Probe and Sensor Metrics (Aggregated by Target) (discovery_perf_metric_probe_sensor_rollup_by_target) table.

These properties control the gathering of probe and sensor metrics:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.perf.metrics.enable_collection</td>
<td>Enables collection of performance metrics.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default</strong>: true</td>
</tr>
<tr>
<td>glide.discovery.perf.metrics.rollup_by_build</td>
<td>Enables aggregation of individual discovery performance metrics into a roll-up table that groups by build.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default</strong>: true</td>
</tr>
<tr>
<td>glide.discovery.perf.metrics.rollup_by_status</td>
<td>Enables aggregation of individual discovery performance metrics into a roll-up table that groups by discovery status.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Type</strong>: true</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default</strong>: false</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.discovery.perf.metrics.rollup_by_target</td>
<td>Enables aggregation of individual discovery performance metrics into a roll-up table that groups by target IP address. By default, Discovery continuously collects individual IP address metrics, even when aggregation by target IP is disabled. When you enable IP target aggregation, Discovery includes all target metrics in the roll-up table.</td>
</tr>
</tbody>
</table>
|                                             | - **Type**: true | false  
|                                             | - **Default**: false                                                                                                                                                                                     |

**View Discovery performance metrics for probes, sensors, and patterns**

By default, Discovery tracks the performance of individual probes, sensors, and patterns by measuring the processing time. When patterns are used, Discovery measures the Identification and Reconciliation Engine (IRE) processing time.

Role required: discovery_admin, admin

1. Navigate to **Discovery > Discovery Performance Metrics > Probe/Sensor (Individual)**.
2. Sort the list by Discovery status to see the list of probes and patterns that ran in a specific Discovery.
3. You can view the metrics for each probe, sensor, or pattern in the list or open the record. All probe and sensor metrics data are read-only.
The Probe and Sensor Metrics (Individual) form provides these fields:

<table>
<thead>
<tr>
<th>Field label</th>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build/version</td>
<td>build_version</td>
<td>Build on which the Discovery was run.</td>
</tr>
<tr>
<td>Discovery status</td>
<td>discovery_status</td>
<td>ID number of the Discovery status from which the metrics were collected.</td>
</tr>
<tr>
<td>Target IP address</td>
<td>target_ip</td>
<td>IP address of the target for this Discovery.</td>
</tr>
<tr>
<td>ECC queue input</td>
<td>ecc_queue_input</td>
<td>Identifies a particular ECC input record in the ECC queue table.</td>
</tr>
<tr>
<td>ECC queue topic *</td>
<td>ecc_queue_topic</td>
<td>Identifies the Java class in the MID Server that executes the probe.</td>
</tr>
<tr>
<td>ECC queue name *</td>
<td>ecc_queue_name</td>
<td>Identifies the probe/pattern evaluated for performance in this aggregation.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field label</th>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe *</td>
<td>probe</td>
<td>Name of the probe used for this Discovery.</td>
</tr>
<tr>
<td>Probe processing time</td>
<td>probe_time</td>
<td>Interaction time of the MID Server with the target, including construction of the payload that is sent back to the instance. The time is in milliseconds.</td>
</tr>
<tr>
<td>IRE processing time</td>
<td>ire_time</td>
<td>Time required to process the pattern payload on the instance by the Identification and Reconciliation Engine (IRE). IRE time is useful because it shows the part of the sensor time used by the pattern. The time is in milliseconds.</td>
</tr>
<tr>
<td>Sensor processing time</td>
<td>sensor_time</td>
<td>Time it took the sensor to process the payload on the instance for a Discovery. The time is in milliseconds.</td>
</tr>
</tbody>
</table>

* Used to uniquely identify a probe/pattern and accompanying sensor when gathering metrics for a Discovery.

*View Discovery performance metrics aggregated by build*

Use the roll-up by build data to ensure that the processing times for Discovery components remain consistent for discoveries in a 24 hour period. View aggregate build data before and after an upgrade to compare the performance of the old and new versions. All aggregated performance data is read-only.

Role required: discovery_admin, admin

1. Navigate to Discovery > Discovery Performance Metrics > Probe/Sensor (Rollup-By-Build).
2. Sort the list by Build/version.
3. Filter by a specific build to see the aggregated processing times for the probes and patterns that performed a Discovery on that build.

4. Open a record to see the probe/pattern statistics for the selected build.
   The form displays additional fields not visible on the list. Roll-up calculations are over a 24 hour period, beginning every night at 0200.

5. See the table of performance aggregation data for descriptions of additional metrics displayed on the form for roll-ups by build.

View Discovery performance metrics aggregated by status
Use the roll-up by status data to ensure that the processing times for probes/patterns and sensors remain consistent for a specific Discovery. All aggregated performance data is read-only.

Role required: discovery_admin, admin

1. Navigate to Discovery > Discovery Performance Metrics > Probe/Sensor (Rollup-By-Status).
2. Sort the list by **Discovery status** to see the aggregated processing times for the probes and patterns that ran during a specific Discovery.

3. Filter by a specific status to display metrics available for probes and patterns that ran in that status.

4. Open a record to see the probe/pattern statistics for the selected status.
   The form displays additional fields not visible on the list. Roll-ups are only created for a completed or cancelled status.

5. See the table of performance aggregation data for descriptions of additional metrics displayed on the form for roll-ups by status.

View Discovery performance metrics aggregated by IP address
Use the roll-up by target data to ensure that the processing times for probes/patterns and sensors remain consistent for each Discovery of a specific IP address. All aggregated performance data is read-only.

Role required: discovery_admin, admin

1. Navigate to **Discovery > Discovery Performance Metrics > Probe/Sensor (Rollup-By-Target).**
2. Sort the list by **Target IP address.**
3. Filter by a specific IP address to see the aggregated processing times for the probes and patterns that performed the Discovery of that IP address.

4. Open a record to see the statistics about the selected probe/pattern for the specific IP address.
   The form displays additional fields not visible on the list. Roll-ups are performed after IP Discovery has completed successfully.

5. See the table of performance aggregation data for descriptions of additional metrics displayed on the form for roll-ups by target.
Aggregated data for Discovery performance metrics
Discovery performance metrics can accumulate data for probes, patterns, and sensors each time Discovery runs. Discovery calculates processing times and increments the number of times a component runs for each roll-up profile: status, target, or build. All aggregated performance data is read-only.

Sample roll-up form

This is an example of an aggregation record for probe and sensor metrics. The metrics fields shown here are used for each aggregation.
<table>
<thead>
<tr>
<th>Discovery status</th>
<th>050019651</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregation cutoff</td>
<td>2018-10-11 13:18:36</td>
</tr>
<tr>
<td>Build/version</td>
<td>glide-trackdisco-03-09-2015_10-10-2018_22</td>
</tr>
<tr>
<td>ECC queue topic</td>
<td>Shazzam</td>
</tr>
<tr>
<td>ECC queue name</td>
<td>npm</td>
</tr>
<tr>
<td>Probe</td>
<td>Shazzam</td>
</tr>
<tr>
<td>Probe time (count)</td>
<td>1</td>
</tr>
<tr>
<td>Probe time (average)</td>
<td>4,107</td>
</tr>
<tr>
<td>Probe time (minimum)</td>
<td>4,107</td>
</tr>
<tr>
<td>Probe time (maximum)</td>
<td>4,107</td>
</tr>
<tr>
<td>Probe time (total)</td>
<td>4,107</td>
</tr>
<tr>
<td>Sensor time (count)</td>
<td>1</td>
</tr>
<tr>
<td>Sensor time (average)</td>
<td>508</td>
</tr>
<tr>
<td>Sensor time (minimum)</td>
<td>508</td>
</tr>
<tr>
<td>Sensor time (maximum)</td>
<td>508</td>
</tr>
<tr>
<td>IRE time (total)</td>
<td>508</td>
</tr>
<tr>
<td>IRE time (count)</td>
<td>1</td>
</tr>
<tr>
<td>IRE time (average)</td>
<td></td>
</tr>
<tr>
<td>IRE time (minimum)</td>
<td></td>
</tr>
<tr>
<td>IRE time (maximum)</td>
<td></td>
</tr>
</tbody>
</table>
## Performance Framework aggregated data

Except where noted, these fields are common to all aggregation records.

<table>
<thead>
<tr>
<th>Field label</th>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build/version</td>
<td>build_version</td>
<td>Name of the build running on the instance. This name identifies the version, patch level, and release date of the ServiceNow platform.</td>
</tr>
<tr>
<td>Discovery status</td>
<td>discovery_status</td>
<td>ID number of the Discovery status record for this aggregation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: This field only appears on the form for roll-ups by status.</td>
</tr>
<tr>
<td>Target IP address</td>
<td>target_ip_address</td>
<td>IP address of the target for this Discovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: This field only appears on the form for roll-ups by target.</td>
</tr>
<tr>
<td>Aggregation cutoff</td>
<td>aggregation_cutoff</td>
<td>The cutoff time varies, depending on the roll-up profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>By-build</strong>: Occurs daily at 02:00, by default.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>By-status</strong>: Closing time of the last aggregation for that Discovery status, which might have run the last time <code>discovery.complete</code> or <code>discovery.cancelled</code> ran for that status.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>By-target</strong>: Closing time of the last aggregation for that target IP address, which might have run the last time <code>discovery.device.complete</code> ran for that IP address.</td>
</tr>
<tr>
<td>ECC queue topic *</td>
<td>ecc_queue_topic</td>
<td>Identifies the Java class in the MID Server that executes the probe.</td>
</tr>
<tr>
<td>ECC queue name *</td>
<td>ecc_queue_name</td>
<td>Identifies the probe/pattern evaluated for performance in this aggregation.</td>
</tr>
<tr>
<td>Probe *</td>
<td>probe</td>
<td>Name of the probe used for this Discovery.</td>
</tr>
<tr>
<td>Probe time (count)</td>
<td>probe_time_count</td>
<td>Number of times a probe ran for a given roll-up profile.</td>
</tr>
<tr>
<td>Field label</td>
<td>Field name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Probe time (average)</td>
<td>probe_time_average</td>
<td>Average time a probe took to gather data on the target and format the payload for a given roll-up profile.</td>
</tr>
<tr>
<td>Probe time (minimum)</td>
<td>probe_time_min</td>
<td>Minimum time a probe took to gather data on the target and format the payload for a given roll-up profile.</td>
</tr>
<tr>
<td>Probe time (maximum)</td>
<td>probe_time_max</td>
<td>Maximum time a probe took to gather data on the target and format the payload for a given roll-up profile.</td>
</tr>
<tr>
<td>Probe time (total)</td>
<td>probe_time_total</td>
<td>Total time used by a probe to gather data on the target and format the payload for a given roll-up profile.</td>
</tr>
<tr>
<td>Sensor time (count)</td>
<td>sensor_time_count</td>
<td>Number of times a sensor processed payloads for a given roll-up profile.</td>
</tr>
<tr>
<td>Sensor time (average)</td>
<td>sensor_time_average</td>
<td>Average time a sensor took to process payloads on the instance for a given roll-up profile.</td>
</tr>
<tr>
<td>Sensor time (minimum)</td>
<td>sensor_time_min</td>
<td>Minimum time a sensor took to process a payload on the instance for a given roll-up profile.</td>
</tr>
<tr>
<td>Sensor time (maximum)</td>
<td>sensor_time_max</td>
<td>Maximum time a sensor took to process a payload on the instance for a given roll-up profile.</td>
</tr>
<tr>
<td>Sensor time (total)</td>
<td>sensor_time_total</td>
<td>Total time used by a sensor to process payloads on the instance for a given roll-up profile.</td>
</tr>
<tr>
<td>IRE time (count)</td>
<td>ire_time_count</td>
<td>Number of times a pattern's payload was processed by the Identification and Reconciliation Engine (IRE) for a given roll-up profile.</td>
</tr>
<tr>
<td>IRE time (average)</td>
<td>ire_time_average</td>
<td>Average time used for IRE processing of a pattern's payload for a given roll-up profile.</td>
</tr>
<tr>
<td>IRE time (minimum)</td>
<td>ire_time_min</td>
<td>Minimum time used for IRE processing of a pattern's payload for a given roll-up profile.</td>
</tr>
<tr>
<td>IRE time (maximum)</td>
<td>ire_time_max</td>
<td>Maximum time used for IRE processing of a pattern's payload for a given roll-up profile.</td>
</tr>
<tr>
<td>Field label</td>
<td>Field name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IRE time (total)</td>
<td>ire_time_total</td>
<td>Total time used for IRE processing of a pattern's payload for a given roll-up profile.</td>
</tr>
</tbody>
</table>

* Together, these values uniquely identify a probe/sensor pair (a "probe execution") that is used for a Discovery.

**Discovery error messages**

Error messages and warnings in the system are documented to allow users to recognize the issues they are having and to take steps to resolve their problems.

**Common error messages**

**Input/output errors**

- CONNECTION_FAILED
- CONNECTION_TIMEOUT
- CONNECTION_REFUSED
- CONNECTION_CIPHER_UNSUPPORTED
- CONNECTION_PROTOCOL_UNSUPPORTED
- CONNECTION_LOST
- INVALID_CONNECTION_PARAMETER
- FILE_NOT_FOUND
- IO_ERROR

**Authentication errors**

- NO_CREDENTIALS
- AUTHENTICATION_FAILED
- INVALID_CREDENTIALS_TYPE

**SSH errors**

- SSH_SHELL_UNSUPPORTED
- SSH_SCRIPT_TRANSFER_FAILED
- SSH_INVALID_SESSION

**Windows errors**

- POWER_SHELL_VERSION_UNSUPPORTED

**Command errors**

- COMMAND_TIMEOUT
- INVALID_COMMAND
- COMMAND_FILTER_ERROR
- COMMAND_PARSER_ERROR
- COMMAND_KEY_MAPPER_ERROR
- COMMAND_VALUE_NORMALIZER_ERROR
- TERMINAL_SESSION_LOST_ERROR
- TERMINAL_SESSION_ENVIRONMENT_MODIFICATION_ERROR

Generic component errors
- INVALID_COMPONENT
- INVALID_COMPONENT_PARAMETER
- INVALID_OPERATION

General error messages

Message: org.xml.sax.SAXParseException: The entity name must immediately follow the ' & ' in the entity reference
A version of this message occurs if you have used special characters in a password that is saved in an XML file.

Message: Identified, ignored extra IP
This message can appear during the identification phase of Discovery if a targeted IP address belongs to a device that is being discovered at the same time.

For example, a Windows server has two NIC cards with two IP addresses. Discovery targets both IP addresses within the same Discovery schedule. This message is generated to note that that second IP address is ignored because we don't want to update the same CI twice within the same Discovery run.

This message is a warning and is expected. No action is needed.

Message: Authentication failures
The discovery process could not discover the CI because the discovery application could not authenticate. To resolve, add the credentials of that machine in to the discovery credentials table.

Message: Identified, not updating CI, now finished
No match on any of the CI Identifiers.

Message: The impersonation of user failed
This message originates in the Powershell. Check that the domain is specified along with the user name in the credentials.

Message: Connection failed to WMI service. Error: Permission denied
This message originates in WMI. Check that the MID Server service is running with the correct credentials and has access to the target device.

To check this, run the following command from the command prompt on the MID Server host:

```
wmic /node:target /user:user /password:password path win32_operatingsystem
```

target = IP address of target device
user = user account used by the mid server service
password = password used by the mid server service

Message: Connection failed to WMI service. Error: The remote server machine does not exist or is unavailable
This message originates in WMI. Check that the MID Server service account has access to the targeted machine. Check if a domain admin account is used as the MID Server service
account. Check if any existing firewalls are open to the connection. To check this, run the following command from the command prompt on the MID Server host.

Execute for runner_type=WMI:

```plaintext
wmic /node:"<target>" /user:"<user>" /password:"<password>" path win32_operatingsystem
```

From within a Powershell console on the mid server host, execute for runner_type=Powershell:

```plaintext
gwmi win32_operatingsystem -computer <ip> -credential '<username>'
```

**Message: Provider is not capable of the attempted operation**

WMI repository was corrupted. After following the "WMI: Repository Corruption, or Not?" article on the Microsoft TechNet site, the problem was corrected.

**Message: The result file can't be fetched because it doesn't exist**

PRB581515 - Powershell does not work when customer has locked down write rights to admin share.

**Message: Please run sneep as root to ensure correct serial number from fserial data source**

The Oracle Sneep command line tool must be installed for the Solaris - Serial Number probe to work correctly. There is a **known limitation** with Fujitsu PRIMEPOWER devices. To work around this limitation, run the Solaris discovery with root credentials.

**Discovery sensor error messages**

**Message: The multisensor will not process because its major version = X while probe_name responding script's major version = Y**

**Message: The multisensor will not process because its responding script's major version = X while its referenced probe probe_name major version = Y**

**Message: sensor_name sensor's major version = X while its related probe's major version = Y**

The above error messages indicate that there is a major version mismatch in the probe and sensor versions and the sensor will stop processing until this condition is resolved.

**Message: sensor_name multisensor's minor version = X while probe_name responding script's minor version = Y**

**Message: sensor_name multisensor's responding script's minor version = X while its referenced probe probe_name minor version = Y**

**Message: sensor_name sensor's minor version = X while its related probe's minor version = Y**

The above error messages indicate that there is a minor version mismatch in the probe and sensor versions. Processing will continue, but you may want to resolve this condition.

**Message: Message: Sensor error when processing . . . : No sensors defined**

Every active probe looks for a corresponding sensor to process the data that is collected by the probe. The No sensors defined message indicates that the corresponding sensor for the probe is missing or inactive.

**Message: Message: Sensor error when processing . . . : typeError: . . .**

This type error message occurs to indicate the sensor has one of the core error constructors in JavaScript.
Message: Sensor error when processing Linux - Network ARP Tables: Exception while running probe post processing script: No XML data
The system displays the No XML data error when the XML sensor processor fails to find the expected XML data in the probe output. During sensor processing, Discovery attempts to retrieve the probe results but finds the probe output empty. Verify that the probe returns the correct output for the sensor to process.

Message: Sensor error when processing Shazzam: Exception while running probe post processing script: Probe not found: null
The Probe not found error occurs when the sensor processor fails to find the probe record supposedly associated with the sensor. During sensor processing, Discovery gets the probe record from the probe cache and stores it for later reference. The "Probe not found" error occurs either when the sys_id of the probe cannot be found or there is an issue with the probe cache.

See Find the cause of a "Probe not found" error for more information.

Find the cause of a sensor error message
Use this procedure to identify a Discovery sensor error.

Role required: admin

This process requires that you identify the following:

- The error.
- The name of the sensor where the error occurred.
- The datestamp of the error.
- The associated stack trace of the error.

1. Navigate to Discovery > Discovery Log.
2. To find the error, filter on the Level name of Error.
   You can also view the error and the date stamp in the Error Report on the Discovery dashboard.
3. Note the Created field of the error record.
   This timestamp is used to find the proper stack trace.
4. Note the Short Message field of the error record.
   This field shows the name of the sensor where the error has occurred. In this example, a generic sensor error occurred in the SMI – Fiber Channel Switch sensor at 4:03 on 10-24-2014.
Error message in fiber channel switch sensor

In this example, a sensor TypeError occurred in the Windows – Installed Software sensor.

Error message in installed software sensor for Windows

5. To find the stack trace, navigate to **System Logs > Errors**.

6. Search for the date and time that matches the value in the **Created** field of the error record in the discovery log.

7. The **Message** field of that entry contains the full stack trace of the error.

The stack trace contains the error message, the sys_id of the script, and the line number where the error occurred. This stack trace shows the following errors:

- **Line 1 and 2**: There was a JavaScript evaluation error on new DiscoverySmiFcSwitchSensor().
- **Line 3**: The Ci.controller variable is undefined.
- **Line 4**: This script_includes line indicates the sysid 7780111 ... d768 and the error occurred at approximately line 26 of the JavaScript file.

Stack trace

8. After determining the error details, you can fix the JavaScript file.
**Identify cause of No Sensor Defined error message**

Every active probe looks for a corresponding sensor to process the data that is collected by the probe. The No Sensors Defined message indicates that the corresponding sensor for the probe is missing or inactive.

**Role required:** admin

To find the cause of this error, identify the **Short Message** field of the error record. This field shows the name of the probe for which the sensor is missing.

1. Navigate to **Discovery > Discovery Log**.
2. Filter on the **Level name** of **Error**.
   
   You can also view the Error Report on the Discovery dashboard.
3. Note the **Short Message** field of the error record.
   
   This field shows the name of the sensor. In this example, the sensor error occurred when the Shazzam probe checked for the existence of the Shazzam sensor.

![Stack trace for No Sensor Defined error](image)

**Fix the cause of a sensor error message**

To fix a Discovery sensor error message, you must fix the JavaScript file containing the code that generated the error.

**Role required:** admin

To fix the JavaScript file:

1. Navigate to the module in Discovery in which the error occurred.
   
   In this example, navigate to **Discovery Definition > Script Includes**.
2. Fix the error.
   
   a) In this example, search for DiscoverySmiFcSwitchSensor in the **Name** field.
Discovery script includes

b) Click on the link to go to the details page for that script include.
c) Modify the JavaScript script on that page to fix the error indicated.
d) Click Update.

Find the cause of a “Probe not found” error
Verify the correct sys_id of the probe associated with the probe parameter in the payload of the ECC Queue input record.

Role required: discovery_admin, admin

1. Record the sys_id of the probe as it appears in the ECC Queue payload.

   ▼<parameters>
   <parameter name="port" value="22"/>
   <parameter name="probe" value="10e0eebd0a0a0b4f61f46a5027df7fb6"/>
   <parameter name="priority" value="0"/>
   <parameter name="use_class" value="discovery_classy_unix"/>

   ECC Queue payload sample

2. Navigate to Discovery Definition > Probes and open the record for the probe you want to inspect.
3. Right-click in the record’s header bar.
4. Select Copy sys_id from the context menu. Follow browser instructions to copy the sys_id if browser security measures restrict this function.
5. Compare this value with the sys_id of the probe from the payload.
   If the sys_id of the probe record does not match the value in the payload, try to determine the cause of the incorrect value.
Out-of-the-box Discovery Performance Analytics Solutions

Performance Analytics Solutions contain preconfigured dashboards. These dashboards contain actionable data visualizations that help you improve your business processes and practices.

Performance Analytics Solutions

Use the Performance Analytics widgets on the dashboard to visualize data over time, analyze your business processes, and identify areas of improvement. With solutions, you can get value from Performance Analytics for your application with minimal setup.

Important: Set up and test Out-of-the-box Performance Analytics Solutions on a sub-production instance before enabling them in production. You can set up and test Performance Analytics on a sub-production instance without a subscription.

Note:
- Solutions include some dashboards that are inactive by default. You can activate these dashboards to make them visible to end users according to your business needs.
- Out-of-the-box solutions and in-form analytics provide all the configuration records required to analyze default applications. Customize these records for use in your production environment.

To enable the solution plugin for Discovery, an admin can navigate to System Definitions > Plugins and activate the Performance Analytics - Content Pack - Discovery plugin.

Discovery patterns used by ITOM Visibility

Service Mapping and Discovery use patterns in their discovery process. The base system contains a wide range of patterns that cover most industry standard network devices and applications. You can customize these patterns and create new ones.

ServiceNow applications refer to devices and applications that comprise an application service as configuration items (CIs).

What discovery patterns are

A pattern is a sequence of steps whose purpose is to detect attributes of a CI and its outbound connections. Service Mapping and Discovery share a set of preconfigured patterns that cover most of the commonly used devices and applications. Patterns can be of infrastructure or application type. Infrastructure patterns are used only by Discovery for creating lists of devices. Application patterns serve both Service Mapping and Discovery that use the same application patterns for their purposes. For example, Discovery runs the horizontal discovery with the Apache Web Server pattern to find and list all Apache Web Servers in your organization. Service Mapping runs the top-down discovery using the same pattern to discover a specific Apache Web Server and place it on an application service map.

Pattern usage by Service Mapping and Discovery

<table>
<thead>
<tr>
<th>Product</th>
<th>Pattern type</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery</td>
<td>Infrastructure pattern</td>
<td>Inventory list of devices</td>
</tr>
</tbody>
</table>
For discovering devices that act as hosts for applications, Service Mapping relies on Discovery. As part of the top-down discovery process, Service Mapping triggers Discovery to perform its horizontal discovery behind the scenes. Service Mapping then uses information on hosts provided by the horizontal discovery to create its application service maps.

Patterns of all types are stored in the Discovery Patterns (sa_pattern) table.

Discovery uses a combination of probes and patterns. For more information, see Horizontal discovery process flow with probes and sensors.

**Correlation between pattern and CI type**

Patterns are assigned to the CI types that they serve to discover. If necessary, you may assign more than one CI type per pattern. In that case, you define one main CI type and multiple related CI types. For example, a pattern for discovering BIG-IP Global Traffic Manager (GTM) F5 has BIG-IP Global Traffic Manager (GTM) F5 as its main CI type and related CI types for the DNS name, network adapter and other components.

For top-down discovery performed by Service Mapping, each application pattern serves to discover only the main CI type.

At the same time, Service Mapping usually uses more than one pattern to discover the same CI type, since a CI type can use different protocols, operating systems, entry points, and so on.

Unlike top-down discovery, the process of horizontal discovery uses each pattern to discover a main CI type with all related CI types.
Why install patterns from the ServiceNow Store

ServiceNow releases new discovery patterns on the ServiceNow Store on a monthly basis to ensure that your organization can discover the latest industry-standard devices and applications. Major ServiceNow versions incorporate patterns previously released on the ServiceNow Store.

Why customize patterns

You can customize patterns in the following cases:

- If your organization uses proprietary devices and applications, create patterns for these items to enable Discovery and Service Mapping to discover them.
- If you modify key attributes of CI types that had corresponding patterns, modify the relevant patterns to reflect the change.

Pattern versions

When you customize a pattern, you actually create a copy of the original pre-configured pattern. While Service Mapping or Discovery use the customized version, the original version is not deleted. When you upgrade your Now Platform, it updates the original pattern, not the customized copy of it.
If at some point you want to abandon the customized pattern and start using the updated original pattern, you can revert to the original pattern as described in [Choose the pattern version](#).

**Who can customize patterns**

Users must have the pd_admin role to customize patterns. In the base system, the sm_admin role contains the pd_role. Customizing patterns requires basic knowledge of programming.

**Patterns for instances using domain separation**

In instances that use domain separation, patterns may be domain-specific, covering only domains that you created them for, or global, applying to all domains.

Patterns belong to domains. By default, all preconfigured patterns are assigned to the global domain and apply to all domains of all levels.

You can create patterns for specific domains. In that case, the new pattern is used only for this domain and does not exist in any other domains. If you customize an existing pattern in the global domain and assign it to a specific domain, you create a copy of the global pattern, which is still used for all other domains except the domain that has the customized version of this pattern. Likewise, if you customize the pattern belonging to the global domain, the change effects all domains except the one that uses a customized copy of this pattern.
Pattern creation or modification flow

Typically, you maintain two ServiceNow instances in your organization: for production and for development. Create or modify patterns, test them, and verify results in the development instance. When you are satisfied with the discovery results, export relevant patterns from the development instance to create an update set. Then you retrieve and commit the update set in your production instance.

If you are creating a pattern for applications and devices that are not supported in the base system, start from creating CI types for them.
Available discovery patterns

ITOM Visibility comes with an extensive library of patterns. Learn about the discovery patterns that became available since the Kingston release.

ServiceNow releases new discovery patterns on the ServiceNow Store on a monthly basis to ensure that your organization can discover the latest industry-standard devices and applications. Major ServiceNow versions incorporate patterns previously released on the ServiceNow Store.

For a complete list of devices and applications that ITOM Visibility can discover, see Applications supported by Discovery and Service Mapping.
Apigee Edge discovery

Discovery finds Apigee Edge Enterprise edition versions 4.x.x using the APIGee pattern.

The APIGee pattern uses a built-in cache mechanism that collects data from each API service into a local cache file. Service Mapping performs top-down discovery on these cache files to find outgoing Apigee Edge connections.

You can use this pattern on the ServiceNow platform of Jakarta release or later.

Prerequisites

Linux user

Provide the Linux operating system (OS) user with elevated rights for running the cat, ls, netstat, and stat commands. For more information, refer to Service Mapping commands requiring a privileged user in Service Mapping documentation.

In addition, provide this Linux user with permissions to run the following commands on the OS user home folder:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;date +%s&quot;</td>
<td>Gets the current date.</td>
</tr>
<tr>
<td>&quot;ls -d $HOME/APIGee_Cache/cache_folder*&quot;</td>
<td>Gets the last cache folder.</td>
</tr>
<tr>
<td>&quot;mkdir-p $HOME/APIGee_Cache/cache_folder_&quot; + $today + &quot;; ls -d $HOME/APIGee_Cache/cache_folder_&quot; + $today</td>
<td>Creates the cache file if the cache file is empty.</td>
</tr>
<tr>
<td>&quot;ls -d $HOME/APIGee_Cache/cache_folder*&quot;</td>
<td>Verifies that the cache file has been created.</td>
</tr>
<tr>
<td>&quot;echo &quot;expr &quot; + $today + &quot; -&quot; + $current_date+ &quot; &quot; \geq &quot; + $week_seconds+ &quot; \; fi&quot;</td>
<td>Calculates the gap between the folder creation time and the current date.</td>
</tr>
<tr>
<td>&quot;ls -d $HOME/APIGee_Cache/cache_folder*&quot;</td>
<td>Checks if existing cache files contain the source URL that you enter as the entry point for the service.</td>
</tr>
<tr>
<td>&quot;ls + $cache_folder+ &quot;/&quot; s ort &quot; xarg sgrep -l &quot; + $entry_point.u_source_url</td>
<td>Checks if it is necessary to recreate the folder.</td>
</tr>
<tr>
<td>&quot;rm -rf $cache_folder+ &quot; s ort &quot; xarg sgrep -l &quot; + $entry_point.u_source_url</td>
<td>Removes old cache files.</td>
</tr>
<tr>
<td>&quot;curl -k -X GET &quot;URL&quot; -H &quot;&lt;ClientID&gt;: &lt;Token&gt;&quot;</td>
<td>Gets the API response for the given token.</td>
</tr>
<tr>
<td>&quot;ls + $cache_folder+ &quot;/&quot; s ort &quot; xarg sgrep -l &quot; + $entry_point.u_source_url</td>
<td>Gets the URL definition from the cache file.</td>
</tr>
</tbody>
</table>

Applicative credential

Configure the applicative credential as follows:

Applicative Credentials form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The credential name, for example Apigee token.</td>
</tr>
<tr>
<td>Active</td>
<td>Check box for enabling this credential for discovery.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>User name</td>
<td>The name of the user of this applicative credential. You can use any user for the credential for this pattern, since the information is extracted from a local cache.</td>
</tr>
<tr>
<td>Password</td>
<td>The actual user password of this applicative credential.</td>
</tr>
<tr>
<td>CI type</td>
<td>The CI type for which this credential is used: APIGee Service (u_cmdb_ci_appl_apigee_srv).</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>ServiceNow applications refer to devices and applications that comprise an application service as configuration items (CIs).</td>
</tr>
<tr>
<td>Applies to</td>
<td>Select whether to apply these credentials to All MID servers in your network, or to one or more Specific MID servers. Specify the MID Servers that should use these credentials in the MID servers field.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter the order (sequence) in which the platform tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order.</td>
</tr>
</tbody>
</table>

**curl command**

Make sure that the curl command is installed on the Linux server hosting Apigee Edge.

**Apigee URLs for discovering connections**

In the first step of the connection section of the APIGee pattern, configure URLs of the outgoing Apigee Edge connections in the following syntax:

```
"01#DVL#https://<url1>.com/dashboard/v1/targetServers/dvl### 02#QA#https://<url2>.com/dashboard/v2/targetServers/dvl###
```

Where

- ‘01’ is the URL number,
- ‘DVL’ is the environment type, like development.
Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apigee Edge (cmdb_ci_appl_apigee_srv)</td>
<td>The name of the server hosting the Apigee Edge.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the server hosting the Apigee Edge.</td>
</tr>
<tr>
<td>Version</td>
<td>The version of the Apigee Edge.</td>
</tr>
<tr>
<td>Installation directory</td>
<td>The folder containing all the Apigee Edge libraries and executable files.</td>
</tr>
<tr>
<td>Configuration directory</td>
<td>The folder containing all the Apigee Edge configuration files.</td>
</tr>
</tbody>
</table>

CI relationships

The APIGee pattern does not create any CI relationships.

Data collected by Service Mapping during top-down discovery

The APIGee pattern identifies connections from the Apigee Edge to other CIs based on the URL you configure in the connection section of the pattern.

Amazon AWS application ELB Service discovery

Discovery uses the Amazon AWS application ELB Service discovery pattern to show all load balancers in your environment in a map.

Elastic Load Balancing (ELB) automatically distributes incoming application traffic across multiple targets.

The Amazon AWS application ELB Service discovery pattern supports top-down discovery. This pattern discovers all load balancers with these characteristics:

- Use ultra-high performance, TLS offloading at scale, and centralized certificate deployment
- Support UDP and static IP addresses for your application

For top-down discovery, the entry point is HTTP(s) Endpoint. Outgoing relations are to Amazon EC2 instances.

You can use this pattern on the ServiceNow platform using Kingston or later releases.
Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

**Prerequisites**

- Run a horizontal discovery, using Cloud API (CAPI). This is necessary because this discovery pattern supports top-down discovery only. For cloud discovery, you can run the horizontal discovery as usual.
- When the load balancers data has been stored in the CMDB, create an application service for each load balancer.
- Set read-only permissions for these REST APIs:

**Data collected by Discovery during horizontal discovery**

The discovered data includes the following tables and fields.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Balancer Services (cmdb_ci_lb_service)</td>
<td>Fully Qualified Domain Name (FQDN) of the load balancer. The DNS name.</td>
</tr>
<tr>
<td>Fully qualified domain name (fqdn)</td>
<td>Name of the load balancer, in the following format: &lt;LB name&gt;:&lt;LB port&gt;.</td>
</tr>
<tr>
<td>Name (name)</td>
<td>Serial number of the load balancer, in the following format: #Object_ID=&lt;LB_ARN&gt;.</td>
</tr>
<tr>
<td>Serial Number (serial_number)</td>
<td>Port of the load balancer.</td>
</tr>
<tr>
<td>Port (port)</td>
<td>Object ID (object_id)</td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>IP Address (ip_address)</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td>Unique identifier of the load balancer.</td>
</tr>
</tbody>
</table>

**Amazon AWS classic ELB Service discovery**

Discovery uses the Amazon AWS classic ELB Service discovery pattern to find all load balancers that use HTTP and HTTPS traffic.

Elastic Load Balancing (ELB) automatically distributes incoming application traffic across multiple targets.
The Amazon AWS classic ELB Service discovery pattern supports top-down discovery. The two possible entry points for top-down discovery are the HTTP(s) Endpoint and TCP Endpoint. Outgoing relations are to Amazon EC2 instances.

You can use this pattern on the ServiceNow platform using Jakarta or later releases.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

**Prerequisites**

- Run a horizontal discovery, using Cloud API (CAPI). This is necessary because this discovery pattern supports top-down discovery only. For cloud discovery, you can run the horizontal discovery as usual.
- When the load balancers data has been stored in the CMDB, create an application service for each load balancer.
- Set read-only permissions for these REST APIs:

**Data collected by Discovery during horizontal discovery**

The discovered data includes the following tables and fields.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Balancer Services (cmdb_ci_lb_service)</td>
<td>Fully Qualified Domain Name (FQDN) of the load balancer. The DNS name.</td>
</tr>
<tr>
<td>Fully qualified domain name (fqdn)</td>
<td>Name of the load balancer, in the following format: &lt;LB name&gt;:&lt;LB port&gt;.</td>
</tr>
<tr>
<td>Name (name)</td>
<td>Port of the instance.</td>
</tr>
<tr>
<td>Back End Port (back_end_port)</td>
<td>Port of the instance.</td>
</tr>
<tr>
<td>Server Port (server_port)</td>
<td>Port of the instance.</td>
</tr>
<tr>
<td>Port (port)</td>
<td>Port of the load balancer.</td>
</tr>
<tr>
<td>Front End Port (front_end_port)</td>
<td>Port of the load balancer.</td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>LB port.</td>
</tr>
<tr>
<td>IP Address (ip_address)</td>
<td>IP address of the FQDN.</td>
</tr>
<tr>
<td>IP Address (cmdb_ci_ip_address)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>Name of the load balancer, in the following format: &lt;LB name&gt;:&lt;LB port&gt;.</td>
</tr>
<tr>
<td>Netmask (netmask)</td>
<td>0.0.0.0</td>
</tr>
</tbody>
</table>
Amazon AWS Relational Database Service discovery

The ServiceNow® Discovery application uses the Amazon AWS Relational Database Service discovery pattern to find AWS RDS components.

Amazon Relational Database Service (RDS) is a managed relational database service by Amazon Web Services (AWS). It enables users to easily create and manage relational database instances in the cloud.

Discovery uses the discovery pattern to run horizontal discovery, events discovery, and tags discovery. The pattern uses a set of Amazon REST APIs to find the RDS components.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

- Install the Discovery and Service Mapping Patterns application from the ServiceNow Store.
- On the Now Platform, configure AWS credentials, using a secret key and an access key.
- Create a service account. Set the Account ID to the Amazon account ID to which RDS belongs. Use the Account ID as it appears in the AWS Management Console.
- Set read-only permissions for these REST APIs:
  - https://rds.<REGION>.amazonaws.com/?Action=DescribeDBInstances
  - https://rds.<REGION>.amazonaws.com/?Action=DescribeDBInstances&DBInstanceIdentifier=<input_object_id> for events discovery
  - https://tagging.<REGION>.amazonaws.com/ for tags discovery
- Ensure to place the MID Server inside the Amazon cloud.

Data collected by Discovery during horizontal discovery

The discovered data includes the following tables and fields.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Database (cmdb_ci_cloud_database)</td>
<td>Class of the database. Includes templates for CPU and memory.</td>
</tr>
<tr>
<td>Category (category)</td>
<td>Class of the database.</td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>Unique resource ID that identifies each logical datacenter (LDC).</td>
</tr>
<tr>
<td>Fully qualified domain name (fqdn)</td>
<td>Address of the database endpoint. The DNS name.</td>
</tr>
<tr>
<td>TCP Port (tcp_port)</td>
<td>Port of the database endpoint.</td>
</tr>
<tr>
<td>Name (name)</td>
<td>Name of the database.</td>
</tr>
<tr>
<td>Version (version)</td>
<td>Version of the database.</td>
</tr>
<tr>
<td>Type (type)</td>
<td>Type of the database. Can be one of the following: Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, SQL Server.</td>
</tr>
<tr>
<td>Operational Status (operational_status)</td>
<td>Status of the database.</td>
</tr>
</tbody>
</table>
Amazon DynamoDB discovery

Discovery and Service Mapping use the Amazon AWS DynamoDB pattern to find components of DynamoDB.

The Amazon AWS DynamoDB pattern deploys Amazon REST APIs for fetching information about DynamoDB. You can also configure Service Mapping and Discovery to discover AWS DynamoDB components using AWS events.

You can use this pattern on the Now Platform® using Jakarta Patch 10, Kingston Patch 8, or London Patch 2.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

User access

On the AWS Console, provide a user with the following permissions: AmazonDynamoDBReadOnly Access and AWSLambdaInvocation-DynamoDB.

Cloud Service Account

On the Now Platform®, set up the service account with these properties:

- An Account ID, which is the Amazon account ID to which DynamoDB belongs. Use the Account ID as it appears in the AWS Management Console.
- The appropriate AWS credentials required for the account ID to reach the AWS cloud.

AWS Credentials

On the Now Platform®, configure AWS credentials, using a secret key and an access key.

Entry point for service containing DynamoDB

Use AWS Resource Name Endpoint as the Entry Point Type when creating a service containing DynamoDB. Provide the Amazon Resource Name (ARN) and host for the entry point.

(For top-down discovery by Service Mapping) Run horizontal discovery of the Logical Datacenters
Discover Logical Datacenters hosting DynamoDBs as described in Run Discovery on an AWS cloud service account in the ServiceNow Cloud Management documentation.

**Discovery schedule**

Create a schedule for the relevant AWS service account as described in Schedule Discovery on a service account in the ServiceNow Cloud Management documentation.

**Configure AWS event service**

To enable alert-based discovery, configure the AWS event service as described in Set up AWS event processing for Discovery and Service Mapping in the ServiceNow Cloud Management documentation.

**Data collected by Discovery during horizontal discovery**

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DynamoDB Global Table (cmdb_ci_dynamodb_global_table)</td>
<td></td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>The Amazon Resource Name (ARN), expressed in this format: arn:aws:dynamodb::&lt;region&gt;::table:&lt;table name&gt;</td>
</tr>
<tr>
<td>DynamoDB Table (cmdb_ci_dynamodb_table)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>Table name.</td>
</tr>
<tr>
<td>Creation date (creation date)</td>
<td>Creation date of DynamoDB according to the time zone of the MID Server that discovers this DynamoDB.</td>
</tr>
<tr>
<td>Cloud provider (cloud_provider)</td>
<td>Amazon AWS</td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>The Amazon Resource Name (ARN), expressed in this format: arn:aws:dynamodb::&lt;region&gt;::table:&lt;table name&gt;</td>
</tr>
<tr>
<td>Global ARN (global_arn)</td>
<td>The global ARN of the table for the global DynamoDB table. This field is empty if the DynamoDB is not global.</td>
</tr>
<tr>
<td>Stream ARN (stream_arn)</td>
<td>The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Stream ARN</td>
</tr>
<tr>
<td></td>
<td>• Disabled</td>
</tr>
<tr>
<td>Service name (service_name)</td>
<td>Set to <strong>DynamoDB</strong> by default.</td>
</tr>
<tr>
<td>Encryption (encryption)</td>
<td>Encryption configuration on DynamoDB. The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Enabled</td>
</tr>
<tr>
<td></td>
<td>• Disabled</td>
</tr>
<tr>
<td>Replication group (replication_group)</td>
<td>Table ARN of the connected global DynamoDB tables.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Read Autoscaling (read_autoscaling)</td>
<td>The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Enabled</td>
</tr>
<tr>
<td></td>
<td>• Disabled</td>
</tr>
<tr>
<td>Read Units (read_units)</td>
<td>The number of read units on the DynamoDB table.</td>
</tr>
<tr>
<td>Read Max Units (read_max_units)</td>
<td>The maximum allowed number of read units on the DynamoDB table.</td>
</tr>
<tr>
<td>Read Min Units (read_min_units)</td>
<td>The minimum allowed number of read units on the DynamoDB table.</td>
</tr>
<tr>
<td>Write Units (write_units)</td>
<td>The number of write units on the DynamoDB table.</td>
</tr>
<tr>
<td>Write Max Units (write_max_units)</td>
<td>The maximum allowed number of write units on the DynamoDB table.</td>
</tr>
<tr>
<td>Write Min Units (write_min_units)</td>
<td>The minimum allowed number of write units on the DynamoDB table.</td>
</tr>
<tr>
<td>Table Size (table_size)</td>
<td>The currently provisioned size of the DynamoDB table in bytes.</td>
</tr>
<tr>
<td>Write Autoscaling (write_autoscaling)</td>
<td>The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Enabled</td>
</tr>
<tr>
<td></td>
<td>• Disabled</td>
</tr>
<tr>
<td>Point-in-Time (point_in_time)</td>
<td>The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Enabled</td>
</tr>
<tr>
<td></td>
<td>• Disabled</td>
</tr>
</tbody>
</table>

The graphic illustrates CIs that are part of DynamoDB discovery.
CI relationships

The Amazon AWS DynamoDB pattern creates the following CI relationships:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>DynamoDB Global Table</td>
<td>Cluster of::Cluster</td>
<td>DynamoDB Table</td>
</tr>
<tr>
<td>(cmdb_ci_dynamodb_global_table)</td>
<td></td>
<td>(cmdb_ci_dynamodb_table)</td>
</tr>
</tbody>
</table>

Data collected by Service Mapping during top-down discovery

The Amazon AWS DynamoDB pattern identifies connections from the DynamoDB to AWS Lambda functions.
Troubleshooting

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout.</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the sa.debugger.max_timeout parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

**Define a Cloud REST Query for cloud computing devices**

As part of creating or modifying a discovery pattern, you can use the Cloud REST Query operation to extract information from configuration items (CIs) of the PaaS (Platform as a Service) type, such as Microsoft Azure or Amazon Web Services.

- Verify that the operating system of the CIs for which you want to use the Cloud REST Query operation extends the Logical Data Center OS type (cmdb_ci_logical_datacenter):
  1. Navigate to Pattern Designer > Discovery Patterns.
  2. Double-click the relevant pattern to open it in the Pattern Designer.
  3. Click the Basic tab.
  4. Note the operating system for this CI.
  5. Navigate to System Definition > Tables.
  6. Set the search field to Label and enter the name of the operating system as stated on the Basic tab of the pattern.
  7. Find the operating system in the list and verify that Logical Datacenter appears in the Extends table column.

- Navigate to the relevant pattern step:
  1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.
  2. Select the relevant pattern step or click the Add a step above icon to add a step.
Basic knowledge of programming is desirable.

Role required: pd_admin

Deploy the Cloud REST Query operation in patterns used for discovery of PaaS CIs. This Java-script based custom operation is available only after downloading patterns version 1.0.24 or later from ServiceNow Store. Use this operation instead of the Cloud REST Call operation available in the base system.

Refer to the official API-related documentation provided by manufacturers to obtain the query syntax for the device you want to query using the Cloud REST Query operation. For example:

- https://docs.microsoft.com/en-us/rest/api/apimanagement/

1. Select **Cloud REST Query** from the **Operation** list.
2. Define query parameters as necessary:

   **Note:** If you customized this operation, the query parameters may be different.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td>Specify the URL as described in the official Microsoft Azure or Amazon Web Services documentation. You can use variables. You can also enter a value from the specific field in a tabular variable.</td>
</tr>
</tbody>
</table>
   | Method    | Enter the method value as a string using all capital letters, for example “GET”. Supported methods are:
   |           | - GET
   |           | - POST
   |           | - PUT
   |           | Refer to the relevant API guide for information. If the API documentation does not specifically mention the HTTP query method, use the GET method. |
   | Body      | (Optional) Enter a request body as a string. If the string contains quotation marks, use the backslash mark in front of the quotation marks to indicate that the string does not end at the quotation marks. |
   | Headers   | (Optional) If the relevant API documentation states that HTTP headers must be sent, enter these headers in the following format:
   |           | Header_name1:header_value1,header_name2:header_value2
   |           | For example, Content-Type:xml. |
   | Formatted | Formatted JSON or not |

3. To save the command output in its entirety as a variable, select **NONE** from the **Define Parsing** list and define the variable.
4. Select the parsing strategy from the **Define Parsing** list.
5. If working in the Debug Mode, define the parsing criteria as follows:
   a) Click Run Operation to see the result in the Output pane.
   b) In the Output pane, mark text or symbols that you want a variable to contain.
   c) In the variable name box, enter the name for the new variable, for example ‘process_name’.
   
   ![Screenshot of parsing criteria]

   d) Press Enter.
   
   The new variable is added in the Variables pane.

6. If you are not working in Debug Mode, define the parsing criteria as follows:
   a) In the Variables pane, click Add Variable and enter the name for the new variable.
   b) Click the Advanced icon.
c) Click **Add Column Label**.

d) Enter the parsing query.

e) Enter a value for the delimiter.

f) Enter a value for the position.

7. Select **Terminate** to stop discovery if no results are found.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Application</td>
</tr>
<tr>
<td>CI Type</td>
<td>Cloud Object Storage (cmdb_ci_cloud_object_storage)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Amazon AWS S3</td>
</tr>
<tr>
<td>Section</td>
<td>Identification of AWS S3</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>3. List buckets</td>
</tr>
</tbody>
</table>
Use the Cloud REST Query operation to extract information about AWS buckets, their names and creation dates.

Amazon Web Services (AWS) Resource Inventory

Discovery and Service Mapping can find and map the AWS resources available by AWS Config service.

Request apps on the Store

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store. For cumulative release notes information for all released apps, refer to the ServiceNow Store version history release notes.
About this pattern

Discovery uses the AWS Resource Inventory pattern to discover the resources available through AWS. Use this pattern for AWS resources which do not have a dedicated pattern, such as AWS Lambda or EC2. This pattern is limited to resources that are available from AWS Config Service, as described in the AWS Config Developer Guide.

You can use this pattern on the ServiceNow platform using London (Patch 8), Madrid (Patch 2), or later releases.

Prerequisites

- Create AWS credentials.
- Create an AWS cloud service account.
  You can use only the AWS master service account for discovery, not its sub-accounts.

  Note: The Cloud Discovery user interface refers to member accounts as sub-accounts.

- Discover AWS Datacenters (LDC)
- Schedule a full AWS discovery or use a specific inventory pattern

- To discover the resources that support the AWS Config service, use API 'https://config.[AWS region].amazonaws.com.' Also, specify the resource type in the API request using the following format: {'resourceType': 'AWS::[RESOURCE]::[TYPE]'}
  The POST method requires the following headers:
  - X-Amz-Target - StarlingDoveService.ListDiscoveredResources
  - Accept - application/json
  - Content-Type - application/x-amz-json-1.1

- Modify the ServiceNow identification rules. Be sure to switch to the Cloud Management Platform application scope before you begin the following procedure.
  1. Navigate to CI Class Manager > Hierarchy > Cloud Resource > Identification Rule.
  2. Delete and add the appropriate identification rule.
     The new identification rule should include only the Object ID attribute.
Modify the ServiceNow related entries

1. Navigate to **Related Entries**.
2. Click **New**.
3. For the identifier, enter `cmdb_ci_cmp_resource`.
4. For the related table, select **Key Value (cmdb_key_value)**.
5. For the referenced field, select **Configuration Item**.
6. For the criterion attributes, add **Key** and **Value**.

You may need to click the lock icon to view and change the criterion attributes.

If your deployment has custom patterns for AWS discovery, ensure that you do not discover AWS resources twice.

1. Ensure that the application scope is Discovery and Service Mapping Patterns:
   a. Navigate to Settings > Developer.
   b. Select Discovery and Service Mapping Patterns from the Application list.

2. Navigate to System Definitions > Tables.
3. Open the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table.
4. Under Related Links, click Show List.
5. Select resource types for which you have custom patterns, and select Delete from the Actions on selected rows list.

The Cloud Inventory Resource Whitelist is predefined with common services. You can expand the whitelist with additional resource types that you want the pattern to discover. The names of these resource types must conform to the appropriate vendor naming conventions.

**Note:** When you modify the out-of-the-box whitelist, it is no longer updated automatically in application updates. You need to maintain the customized list on your own.

1. Open the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table.
2. Click New.
3. Fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Vendor</td>
<td>The vendor of the resource type: AWS.</td>
</tr>
<tr>
<td>Resource Type</td>
<td>The AWS resource type value. For example, AWS::CloudWatch::Alarm.</td>
</tr>
</tbody>
</table>
The changes are applied the next time you run the pattern.

- To discover the resources that support the AWS Config service, use API `https://config.(AWS region).amazonaws.com` Also, specify the resource type in the API request using the following format: `{"resourceType":"AWS::(RESOURCE)::(TYPE)"}`

The POST method requires the following headers:
- X-Amz-Target - StarlingDoveService.ListDiscoveredResources
- Accept - application/json
- Content-Type - application/x-amz-json-1.1

- To discover the resource tags for those resources that support tags, use API `https://tagging.(AWS region).amazonaws.com` Also, specify the resource type filter in the API request using the following format: `{"ResourceTypeFilters": [{"resource"}:"(type)"]}`

The POST method requires the following headers:
- X-Amz-Target - ResourceGroupsTaggingAPI_20170126.GetResources
- Accept - application/json
- Content-Type - application/x-amz-json-1.1

### Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI (cmdb_ci_cmp_resource)</td>
<td>The ID of the item. This is typically the Amazon Resource Name (ARN).</td>
</tr>
<tr>
<td>object_id</td>
<td>Name of the resource.</td>
</tr>
<tr>
<td>resource_type</td>
<td>The asset resource type, based on the contents of the JSON file.</td>
</tr>
<tr>
<td>description</td>
<td>Short description of how the CI is populated.</td>
</tr>
</tbody>
</table>

This pattern extends the cmdb_ci_vm_object as shown below:

- Identifier: None. Inherits identifier from cmdb_ci_vm_object.
- Hosting rules:
  - Hosted on Cloud Service Account
  - Hosted on Logical Datacenter
- Deletion strategy: Marks the CIs as absent.

This pattern populates the cmdb_key_value table with the following data:

- object_id: The ID of the item.
- key: Key in the key-value pair.
• value: Value in the key-value pair.

• The identifier for the related entry of the cmdb_key_value table is inherited from cmdb_ci_vm_object (the parent). The attributes are "key" and "value."

• The deletion strategy can't be applied to the cmdb_key_value table.

CI relationships

The AWS Resource Inventory pattern creates the following CI relationship:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Resource</td>
<td>Hosted on :: Hosts</td>
<td>Logical Datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_cmp_resource)</td>
<td></td>
<td>(cmdb_ci_logical_datacenter)</td>
</tr>
</tbody>
</table>

The following is a dependency view of the collected data flow.
AWS Resource Inventory dependency view

**AWS API gateway discovery**

Discovery and Service Mapping can find AWS API gateways and connections to other entities. Discovery and Service Mapping use the **Amazon AWS API Gateway** pattern to run horizontal and top-down discovery.

You can use this pattern on the ServiceNow platform using the Kingston release or later.

**Prerequisites**

**AWS service account with credentials**

On the Now Platform®, set up the service account with the following:
- An **Account ID**, which is the Amazon account ID to which API Gateway belongs. Use the Account ID as it appears in the AWS Management Console. You can run discovery from the service account form.
- The appropriate AWS credentials required for the account ID to reach the AWS cloud.

Refer to Service accounts for Discovery and Service Mapping in product documentation.

**Update set**

If your instance is on the Jakarta or Istanbul versions, import the update set provided in KB0679927: AWS Lambda top-down discovery update set. This update set is required for Service Mapping.

**User access**

On the AWS Console, provide a user with permissions to run the `apigateway:GET` function for just the GET action in API Gateway.

**Data collected by Discovery during horizontal discovery**

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Gateway (cmdb_ci_cloud_gateway)</td>
<td>The ID of the gateway.</td>
</tr>
<tr>
<td>Object ID</td>
<td>The ID of the gateway.</td>
</tr>
<tr>
<td>Short description</td>
<td>The short description of the gateway.</td>
</tr>
<tr>
<td>DNS Name (cmdb_ci_dns_name)</td>
<td>The unique name or IP address of the DNS host.</td>
</tr>
<tr>
<td>Name</td>
<td>The IP address of the DNS host.</td>
</tr>
<tr>
<td>IP address</td>
<td>The hostname or IP address of the endpoint.</td>
</tr>
<tr>
<td>AWS Resource Name Endpoint (cmdb_ci_endpoint_arn)</td>
<td>The ID (Amazon Resource Name) of other AWS entities to which the gateway connects.</td>
</tr>
</tbody>
</table>

**Configuration item (CI) Relationships**

These relationships are created to support AWS API Gateway discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Gateway (cmdb_ci_cloud_gateway)</td>
<td>Uses:Used by</td>
<td>DNS Name (cmdb_ci_dns_name)</td>
</tr>
</tbody>
</table>
Connections discovered by Service Mapping during the top-down discovery

Service Mapping performs the top-down discovery of the AWS API Gateways in the context of application services. It discovers outgoing connections of the AWS API Gateways:

- AWS Lambda
- VPC Link for Virtual Private Cloud
- HTTP endpoint defined as the configured URL on the API Gateway

Troubleshooting

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000</td>
</tr>
<tr>
<td>an error caused by the REST timeout.</td>
<td>call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>milliseconds. Increase the value of this parameter on the relevant MID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Server and run discovery again.</td>
</tr>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the sa.debugger.max_timeout parameter is set to 240 seconds.</td>
</tr>
<tr>
<td>message contains information about an error caused by a timeout.</td>
<td></td>
<td>Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

Note: If the Configuration Parameters related list for the relevant MID Server does not show this parameter, you may need to add it.
Test patterns for discovering AWS API Gateway and AWS Lambda

Run the horizontal and top-down discovery using Amazon AWS API Gateway (cmdb_ci_cloud_gateway) and Amazon AWS Lambda (cmdb_ci_cloud_function) patterns. Verify that the result is as expected.

Make sure that your deployment complies with all the prerequisites for AWS API gateway discovery and AWS Lambda discovery.

Role required: admin

1. Create a business service using the following properties for its entry point.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI type</td>
<td>HTTPS</td>
</tr>
<tr>
<td>URL</td>
<td>The URL for invoking the API Gateway. Typically, the API GW has the following format: https://{restapi_id}.execute-api.{region}.amazonaws.com/{stage_name}/ For example, <a href="https://k40o0l3af6.execute-api.us-east-1.amazonaws.com/prod/">https://k40o0l3af6.execute-api.us-east-1.amazonaws.com/prod/</a>. For information about the URL format, see <a href="https://docs.aws.amazon.com/apigateway/latest/developerguide/how-to-call-api.html">https://docs.aws.amazon.com/apigateway/latest/developerguide/how-to-call-api.html</a>.</td>
</tr>
</tbody>
</table>

Note: The default stage_name parameter is prod.

2. Click Discover.

Service Mapping runs the top-down discovery using the Amazon AWS API Gateway (cmdb_ci_cloud_gateway) pattern.

After Service Mapping completes identifying the API Gateway configuration item (CI), it runs the Connection to APIs connection section and discovers connection to the AWS Lambda. It then runs the Amazon AWS Lambda (cmdb_ci_cloud_function) pattern and identifies the AWS Lambda CI.

3. After Service Mapping finishes to discover this business service, verify that it discovered and identified AWS components correctly. Check that

- Service Mapping correctly resolves the URL, finds all IP addresses behind it and creates an entry point for each IP address.
Service Mapping identifies the AWS API Gateway and creates an application cluster for IP address.

- The Properties pane shows the properties for the AWS API Gateway: Cloud Gateway (cmdb_ci_cloud_gateway).
- Service Mapping discovers the AWS Lambda connected to the application cluster for API Gateway.
- Service Mapping discovers the databases connected to AWS Lambda.

**AWS Cognito discovery**

Discovery and Service Mapping use the Amazon AWS Cognito pattern to provide authentication, authorization, and user management functions for AWS customers.

You can use this pattern on the Now Platform using London Patch 8, Madrid Patch 2, or later releases.

**Prerequisites**

**User permissions**

Provide user with read-only permission to run the following API:

- `https://cognito-idp.<region>.amazonaws.com`
- Method: POST
- Body: `{"MaxResults": 10}`
- Headers: X-Amz-Target:AWSCognitoIdentityProviderService.ListUserPools,Content-Type:application/x-amz-json-1.0

**AWS Credentials**

On your instance, configure credentials of type AWS Credentials and set to Active.

**Cloud service account**

On your instance, configure the cloud service account of type AWS Datacenter and set to AWS account ID. Use the credentials defined in the preceding AWS Credentials.

**Discovery schedule**

Create a cloud application schedule for discovering AWS Cognito and configure the attributes. Set Discovery to Cloud application.

**Execution pattern**

Create and define the serverless execution pattern for cloud application discovery.

1. Create new Cloud Execution Patterns.
2. Define Name.
3. Verify that Active is true.
4. Verify that Domain is global.
5. Choose the AWS pattern you want to run.
6. Create multiple records if you want to run more than one pattern.

**Discovery schedule for full AWS discovery**

Create a discovery schedule from your Cloud service account created in the earlier procedure.

1. Click on Discover Datacenter and wait for it to finish.
2. Click Create Discovery Schedule.
3. This new schedule is created under the Discovery Schedule and runs all AWS patterns.

**Data collected by Discovery and Service Mapping during horizontal and top-down discovery**

The AWS Cognito pattern collects data.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main C1: cmdb_ci_cloud_authentication</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>A descriptive name used to identify the user pool.</td>
</tr>
<tr>
<td>object_id</td>
<td>This is equal to the account_id and used by IRE identification rules.</td>
</tr>
<tr>
<td>Fqdn</td>
<td>Example of an ARN: arn:aws:cognito-idp:eu-west-1:1751200741520:userpool/eu-west-1_fim5E2mix</td>
</tr>
</tbody>
</table>
Tags are also being collected by an extension section that runs following the pattern. The tagging API for AWS specifies the resource type Cognito.

**Collected information from the AWS Cognito tags**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_key_value</td>
<td></td>
</tr>
<tr>
<td>key</td>
<td>The actual tag key.</td>
</tr>
<tr>
<td>value</td>
<td>The tag value.</td>
</tr>
<tr>
<td>configuration_item</td>
<td>The unique resource ID (ARN) that identifies the resource in the AWS console.</td>
</tr>
</tbody>
</table>

**CI relationships**

The AWS Cognito pattern creates the following CI relationship.

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud authentication</td>
<td>Hosts:Hosted on</td>
<td>Logical datacenter</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_authentication)</td>
<td></td>
<td>(cmdb_ci_logical_datacenter)</td>
</tr>
</tbody>
</table>

**Troubleshooting**

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the <code>mid.sa.cloud.request_timeout</code> parameter.</td>
<td>By default, the <code>mid.sa.cloud.request_timeout</code> parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout.</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the <code>sa.debugger.max_timeout</code> parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

*Note: If the Configuration Parameters related list for the relevant MID Server does not show this parameter, you may need to add it.*
AWS ECS-Fargate resource discovery

Discovery uses the AWS ECS-Fargate pattern to find resources that are managed by the AWS Elastic Container Service (ECS). This includes resources that are run on the Fargate launch type.

AWS ECS is a highly scalable, fast, container management service that makes it easy to run, stop, and manage Docker containers. AWS Fargate is a compute engine for AWS ECS that allows AWS customers to run containers without having to manage servers or clusters.

Discovery uses the AWS ECS-Fargate pattern to run horizontal discovery. You can use this pattern on the ServiceNow platform using Madrid release patch 10 or later.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

- In the AWS console, create AWS credentials with privileges to view ECS resources. For more information, see: https://docs.aws.amazon.com/en_pv/AmazonECS/latest/userguide/get-set-up-for-amazon-ecs.html.

  Note: The API requests use AWS Signature Version 4 to authenticate. When the correct credentials are provided, authentication is done internally for the Cloud API Call pattern steps.

- In Now Platform, create AWS Cloud Service account and credential records. For more information, see Cloud credentials.
- In the Cloud Service Account record, click Discover Datacenters to find AWS Datacenters (LDCs) for the configured AWS Cloud Service account.
- Schedule for either full AWS discovery or only ECS pattern discovery. For more information, see Schedule a horizontal discovery.
- To enable the pattern to find the resource Tags, the parent ECS table (VM Object) must have a related entry for the cmdb_key_value table:
  1. In the Now Platform navigation bar, navigate to Configuration > Identification/Reconciliation, and click CI Identifiers.
  2. Search for and open the VM Object record.
  3. Click the Related Entries tab and make sure it is configured as shown in the VM Object Related Entry figure. If it is not, specify a new related entry: In the Related Entries related list, click New, fill out the form, and then click Submit.
**Data collected by Discovery during horizontal discovery**

This pattern discovers data that provides visibility for ECS clusters, ECS services, ECS task definitions, ECS tasks, Docker containers, and ECS container instances in your organization. The discovered data includes the following tables and fields.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AWS Cloud ECS Cluster (cmdb_ci_cloud_ecs_cluster)</strong></td>
<td></td>
</tr>
<tr>
<td>object_id</td>
<td>The ARN that identifies the cluster.</td>
</tr>
<tr>
<td>name</td>
<td>A user-generated string used to identify the cluster.</td>
</tr>
<tr>
<td>status</td>
<td>The status of the cluster. Valid values: ACTIVE, INACTIVE.</td>
</tr>
<tr>
<td>registered_container_instances</td>
<td>The number of container instances registered to the cluster. The status of these container instances can be either ACTIVE or DRAINING.</td>
</tr>
<tr>
<td><strong>AWS Cloud ECS TaskDefinition (cmdb_ci_cloud_ecs_task_definition)</strong></td>
<td></td>
</tr>
<tr>
<td>object_id</td>
<td>The full ARN of the task definition.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the family to which this task definition is registered.</td>
</tr>
<tr>
<td>status</td>
<td>The status of the task definition.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>revision</td>
<td>The revision of the task in a particular family. This is the version number of the task definition in a family.</td>
</tr>
<tr>
<td>network_mode</td>
<td>The Docker networking mode to use for the containers in the task. Valid values: None, bridge, awsvpc, host.</td>
</tr>
<tr>
<td>cpu</td>
<td>The number of CPU units used by the task.</td>
</tr>
<tr>
<td>memory</td>
<td>The amount of memory, in MiB, used by the task.</td>
</tr>
</tbody>
</table>

AWS Cloud ECS Task (cmdb_ci_cloud_ecs_task)

| object_id            | The ARN that identifies the task.                                           |
| name                 | A user-generated string used to identify the task.                           |
| status               | The last-known status of the task.                                           |
| launch_type          | The launch type on which the task runs.                                      |
| operation_status     | The health status of the task, determined by the health of the essential containers in it. |
| cpu                  | The number of CPU units used by the task, as stated in the task definition. |
| memory               | The amount of memory, in MiB, used by the task, as stated in the task definition. |

AWS Cloud ECS Service (cmdb_ci_cloud_ecs_service)

| object_id            | The ARN that identifies the service.                                         |
| name                 | The name of the service.                                                     |
| status               | The status of the service.                                                   |
| launch_type          | The launch type on which the service runs.                                    |
| scheduling_strategy  | The scheduling strategy to use for the service.                              |

Docker Container (cmdb_ci_docker_container)

| container_id         | The ARN of the container.                                                    |
| name                 | The name of the container.                                                   |
| status               | The last-known status of the container.                                      |
| operational_status   | The health status of the container.                                          |
| cpu                  | The number of CPU units set for the container.                               |
| memory               | The hard limit of memory, in MiB, set for the container.                     |

Virtual Machine Instance (cmdb_ci_vm_instance)

| object_id            | The ARN that identifies the VM instance.                                     |

**Note:** Complete Virtual Machine Instance information is discovered in a different EC2 cloud pattern. In AWS ECS-Fargate resource discovery, only the object_id is parsed to aid with the unique identification of the CI.

<table>
<thead>
<tr>
<th>Key Value (cmdb_key_value)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>key</td>
<td>The Tag key.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>The Tag value.</td>
</tr>
</tbody>
</table>

On the Dependency Views map you can see all discovered resources in your organization that are managed by AWS EWS, and the relationships between them.

In this example, the AWS Cloud ECS Cluster CI is hosted on an AWS Datacenter. It runs two AWS Cloud ECS Tasks and two AWS Cloud ECS Services, and uses three Docker Containers.

**Dependency Views sample map**

**CI relationships**

These relationships are created to support AWS ECS-Fargate discovery:
<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Cloud ECS Cluster</td>
<td>Uses::Used by</td>
<td>Virtual Machine Instance</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
<td></td>
<td>(cmdb_ci_vm_instance)</td>
</tr>
<tr>
<td>AWS Cloud ECS Cluster</td>
<td>Runs::Runs on</td>
<td>AWS Cloud ECS Service</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_service)</td>
</tr>
<tr>
<td>AWS Cloud ECS Cluster</td>
<td>Runs::Runs on</td>
<td>AWS Cloud ECS Task</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_task)</td>
</tr>
<tr>
<td>AWS Cloud ECS Cluster</td>
<td>Hosted on::Hosts</td>
<td>AWS datacenters</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWS Cloud ECS Cluster</td>
<td>Extends::Extended by</td>
<td>Virtual Machine Object</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
<td></td>
<td>(cmdb_ci_vm_object)</td>
</tr>
<tr>
<td>AWS Cloud ECS TaskDefinition</td>
<td>Used by::Uses</td>
<td>AWS Cloud ECS Service</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task_definition)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_service)</td>
</tr>
<tr>
<td>AWS Cloud ECS TaskDefinition</td>
<td>Used by::Uses</td>
<td>AWS Cloud ECS Task</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task_definition)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_task)</td>
</tr>
<tr>
<td>AWS Cloud ECS TaskDefinition</td>
<td>Hosted on::Hosts</td>
<td>AWS Datacenters</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task_definition)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWS Cloud ECS TaskDefinition</td>
<td>Extends::Extended by</td>
<td>Virtual Machine Object</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task_definition)</td>
<td></td>
<td>(cmdb_ci_vm_object)</td>
</tr>
<tr>
<td>AWS Cloud ECS Task</td>
<td>Runs on:: Runs</td>
<td>Virtual Machine Instance</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task)</td>
<td></td>
<td>(cmdb_ci_vm_instance)</td>
</tr>
<tr>
<td>AWS Cloud ECS Task</td>
<td>Runs on:: Runs</td>
<td>Docker Container</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task)</td>
<td></td>
<td>(cmdb_ci_docker_container)</td>
</tr>
<tr>
<td>AWS Cloud ECS Task</td>
<td>Uses::Used by</td>
<td>AWS Cloud ECS TaskDefinition</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_task_definition)</td>
</tr>
<tr>
<td>AWS Cloud ECS Task</td>
<td>Hosted on::Hosts</td>
<td>AWS Datacenters</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWS Cloud ECS Task</td>
<td>Runs on:: Runs</td>
<td>AWS Cloud ECS Cluster</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
</tr>
<tr>
<td>AWS Cloud ECS Task</td>
<td>Extends::Extended by</td>
<td>Virtual Machine Object</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_task)</td>
<td></td>
<td>(cmdb_ci_vm_object)</td>
</tr>
<tr>
<td>AWS Cloud ECS Service</td>
<td>Uses::Used by</td>
<td>AWS Cloud ECS TaskDefinition</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_service)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_task_definition)</td>
</tr>
<tr>
<td>AWS Cloud ECS Service</td>
<td>Hosted on::Hosts</td>
<td>AWS Datacenters</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_service)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWS Cloud ECS Service</td>
<td>Runs on:: Runs</td>
<td>AWS Cloud ECS Cluster</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_service)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
</tr>
<tr>
<td>AWS Cloud ECS Service</td>
<td>Extends::Extended by</td>
<td>Virtual Machine Object</td>
</tr>
<tr>
<td>(cmdb_ci_cloud_ecs_service)</td>
<td></td>
<td>(cmdb_ci_vm_object)</td>
</tr>
<tr>
<td>Docker Container</td>
<td>Hosted on::Hosts</td>
<td>AWS Datacenters</td>
</tr>
<tr>
<td>(cmdb_ci_docker_container)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Docker Container</td>
<td>Runs:: Runs on</td>
<td>AWS Cloud ECS Task</td>
</tr>
<tr>
<td>(cmdb_ci_docker_container)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_task)</td>
</tr>
<tr>
<td>Docker Container</td>
<td>Used by::Uses</td>
<td>AWS Cloud ECS Cluster</td>
</tr>
<tr>
<td>(cmdb_ci_docker_container)</td>
<td></td>
<td>(cmdb_ci_cloud_ecs_cluster)</td>
</tr>
</tbody>
</table>
AWS Lambda discovery

Discovery and Service Mapping can find and map Lambda functions that run in your AWS cloud. Discovery and Service Mapping use the Amazon AWS Lambda pattern to run horizontal and top-down discovery. When you run cloud discovery, your instance queries your AWS service account using the necessary credentials, and then runs a query to obtain Lambda information. The pattern parses this information and then it is saved in the CMDB.

You can use this pattern on the ServiceNow platform using Jakarta Patch 10, Kingston Patch 8, or London Patch 1.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

Update set

If your instance is on the Jakarta or Istanbul versions, import the update set provided in KB0679927: AWS Lambda top-down discovery update set. This update set is required for Service Mapping.

User policy for AWS user

If you use Identity and Access Management (IAM) to manage users in Amazon Web Services (AWS) platform, ensure that you created a user policy for the AWS user. For more information, refer to Create an IAM user policy for Cloud Management in the ServiceNow Cloud Management documentation.

AWS service account with credentials

You must set up your service account with the appropriate AWS credentials. You run discovery from the service account form.

Refer to Service accounts for Discovery and Service Mapping in product documentation.

Service account

On the ServiceNow platform, create a service account. Set the Account ID to the Amazon account IT to which AWS Lambda belongs. Use the Account ID as it appears in the AWS Management Console. Ensure to place the MID Server on the virtual machine that is part of the service account that you are planning to discover.

Discover logical datacenters using horizontal discovery

Click Discover datacenters under Related Links on the Cloud Service Account form.

User permissions

Provide a user with permissions to run the ListFunctions API with the Resource: arn:aws:lambda:region:account-id: function:function-name. Provide a user with permissions to run the GetFunction API with the Resource: `*`.

Amazon Resource Name (ARN)

(Optional for application services using ARN as an entry point) Ensure that you know the ARN for the AWS Lambda you need to discover. ARNs serve as the resource identifiers for Amazon Web Services.

Host

If your instance is on the Jakarta or Istanbul versions, use the following host: 127.0.0.1.
Data collected by Discovery during horizontal discovery

Horizontal discovery populates this data in the CMDB:

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Function (cmdb_ci_cloud_function)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td>The name of the function.</td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>The ID (Amazon Resource Name) of the AWS Lambda function, for example: arn:aws:lambda: {datacenter}1:751200741520:function: {function-name}</td>
</tr>
<tr>
<td>Version (version)</td>
<td>The version of the function. For example, $Latest represents the most recent published version.</td>
</tr>
<tr>
<td>Language (language)</td>
<td>The runtime environment, or language and version, that the function is written in, for example: Node.js6.10 for JavaScript version 6.10.</td>
</tr>
<tr>
<td>Code Location URL (code_location_url)</td>
<td>The URL of the function that you built from the API Gateway console.</td>
</tr>
<tr>
<td>CodeSha256 (codesha256)</td>
<td>The cryptographic hash signature for the function.</td>
</tr>
<tr>
<td>Code Size (code_size)</td>
<td>The size of the function code in MB.</td>
</tr>
<tr>
<td>Function Last Modified (function_last_modified)</td>
<td>The date the function last changed in your AWS Lambda environment.</td>
</tr>
</tbody>
</table>

Note: To see all of these fields, you need to configure the form layout as described in product documentation.

CI relationships

These relationships are created to support AWS Lambda discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Function (cmdb_ci_cloud_function)</td>
<td>Uses:Used by</td>
<td>Cloud Database (cmdb_ci_cloud_database)</td>
</tr>
<tr>
<td></td>
<td>Hosted on:Hosts</td>
<td>AWS Datacenter (cmdb_ci_aws_datacenter)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The AWS datacenter has relationships to other AWS CIs, such as availability zones, service accounts, and so on. Refer to AWS and Azure cloud discovery in product documentation.</td>
</tr>
</tbody>
</table>
### Connections discovered by Service Mapping during the top-down discovery

Service Mapping performs the top-down discovery of the AWS Lambda in the context of application services. It discovers outgoing connections of the AWS Lambda:

- AWS API Gateway
- Amazon Relational Database Service (RDS)
- Other instance of AWS Lambda inside or outside the current region using ARN

### Troubleshooting

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
</tbody>
</table>

**Note:** If the Configuration Parameters related list for the relevant MID Server does not show this parameter, you may need to add it.

| Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout. | The pattern designer fails because of a timeout during pattern debugging (and not during discovery). | By default, the sa.debugger.max_timeout parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server. |

### AWS S3 discovery

The Amazon AWS S3 pattern uses a set of REST API calls to find public and non-public storage buckets of Amazon Simple Storage Service.

In addition to the discovery schedule, the ServiceNow instance triggers the Amazon AWS S3 pattern when AWS Config sends an event to the ServiceNow instance. Config is a service that
ServiceNow continuously monitors AWS resources and sends events to the ServiceNow every time it senses that a resource configuration has changed.

Amazon AWS S3 pattern can also discover AWS S3 buckets located in the AWS GovCloud (US) region. For AWS S3 located in the AWS GovCloud (SU), the pattern cannot discover AWS S3 outgoing connections to other cloud resources like Lambda function. Discovery of outgoing connections for AWS S3 will be supported in future releases.

You can use this pattern on the ServiceNow platform using Jakarta Patch 10, Kingston Patch 8, or London Patch 1.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

- If you use Identity and Access Management (IAM) to manage users in Amazon Web Services (AWS) platform, ensure that you created a user policy for the AWS user. For more information, refer to Create an IAM user policy for Cloud Management in the ServiceNow Cloud Management documentation.
- Configure the AWS service account.
- Configure AWS credentials, using a secret key and an access key.
- AmazonS3ReadOnlyAccess policy user permissions.
- Discover Logical Datacenters hosting S3 buckets as described in Run Discovery on an AWS cloud service account in the ServiceNow Cloud Management documentation.
- Create a schedule for the relevant AWS service account as described in Schedule Discovery on a service account in the ServiceNow Cloud Management documentation.
- To enable alert-based discovery, configure the AWS event service as described in Set up AWS event processing for Discovery and Service Mapping in the ServiceNow Cloud Management documentation.

Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Object Storage (cmdb_ci_cloud_object_storage)</td>
<td></td>
</tr>
<tr>
<td>Bucket name (bucket_name)</td>
<td>The name of the AWS S3 bucket. You can discover all buckets hosted on a Logical Datacenter during one discovery process.</td>
</tr>
<tr>
<td>Creation date (creation date)</td>
<td>Creation date of the AWS S3 bucket.</td>
</tr>
<tr>
<td>Cloud provider (cloud_provider)</td>
<td>Amazon AWS</td>
</tr>
<tr>
<td>Object ID (object_id)</td>
<td>The Amazon Resource Name (ARN), expressed in this format: arn:aws:s3:::&lt;bucket name&gt;</td>
</tr>
<tr>
<td>Service name (service_name)</td>
<td>Set to S3 by default.</td>
</tr>
<tr>
<td>Owner (owner)</td>
<td>Name of the AWS S3 bucket owner.</td>
</tr>
<tr>
<td>Encryption type (encryption_type)</td>
<td>Type of encryption for this bucket. The choices are:</td>
</tr>
<tr>
<td></td>
<td>• None</td>
</tr>
<tr>
<td></td>
<td>• AES-256</td>
</tr>
<tr>
<td></td>
<td>• AWS-KMS</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ACL access type (acl_access_type)</td>
<td>The choices for access control types are:</td>
</tr>
<tr>
<td></td>
<td>• Public</td>
</tr>
<tr>
<td></td>
<td>• Not Public</td>
</tr>
<tr>
<td>Policy access type (policy_access_type)</td>
<td>The choices for policy access types are:</td>
</tr>
<tr>
<td></td>
<td>• Public</td>
</tr>
<tr>
<td></td>
<td>• Not Public</td>
</tr>
<tr>
<td>Replication destination (replication_destination)</td>
<td>Bucket ARN of the Replication Destination bucket.</td>
</tr>
</tbody>
</table>

On the Dependency Views map you can see all discovered S3 buckets in your organization, including S3 buckets that you configured as replicas. In this example, Cloud Object Storage 1, Cloud Object Storage 2, and Cloud Object Storage 3 replicate Cloud Object Storage 2. In its turn, Cloud Object Storage 2 replicates Cloud Object Storage 1.
**CI relationships**

These relationships are created to support AWS S3 discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_cloud_object_storage</td>
<td>Replicates to::Replicated By</td>
<td>cmdb_ci_cloud_object_storage</td>
</tr>
<tr>
<td>cmdb_ci_cloud_object_storage</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_logical_datacenter</td>
</tr>
</tbody>
</table>

**Data discovered by Service Mapping during top-down discovery**

Service Mapping can discover AWS S3 outgoing connections to cloud resources, for example Lambda function.
**Troubleshooting**

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
</tbody>
</table>

*Note:* If the Configuration Parameters related list for the relevant MID Server does not show this parameter, you may need to add it.
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout.</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the <code>sa.debugger.max_timeout</code> parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

**AWS tag discovery**

Discovery and Service Mapping can use patterns to discover the tags for AWS resources and configuration item (CI) types, and then populate the CMDB with these discoveries.

This tag-specific discovery pattern is in addition to the other AWS patterns available on the ServiceNow Store. This pattern extends the ones used for discovering the respective AWS resources.

You can use this pattern to discover tags on the Now Platform® using New York. Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

**Prerequisites**

**AWS Credentials**

- On the Now Platform, configure AWS credentials, using a secret key and an access key.
- Have an Amazon **Account ID**. Use the Account ID as it appears in the AWS Management Console. You can run Discovery from the service account form.
- Set read-only permissions for the following URL: `https://tagging.<region>.amazonaws.com`.

**Discovery schedule**

Create a schedule for the relevant AWS service account as described in Schedule Discovery on a service account in the ServiceNow® Cloud Management documentation.

The CMDB is populated with tags for the following AWS resources and configuration item (CI) types:

- **API Gateway** (`cmdb_ci_cloud_gateway`)
- **Cognito** (`cmdb_ci_cloud_authentication`)
- **DynamoDB** (`cmdb_ci_dynamodb_table`)
- **Lambda** (`cmdb_ci_cloud_function`)
- **RDS** (Relational Database Service) (`cmdb_ci_cloud_database`)
- **S3** (`cmdb_ci_cloud_object_storage`)

**Note:** For more information about the tags and patterns for each of these resources, see the documentation for the specific AWS resource.

When discovering AWS tags, the pattern completes the following high-level tasks:

- Sets the Resource Type filter: Filters the tags to only those tags related to the pattern CI type.
- Initiates the Cloud API call: The API call to the cloud environment that collects the tags and parses their attributes in the Tags table.
- Parses the resource name from the Amazon Resource Name (ARN). Each Amazon resource is assigned a name and a unique identifier.
Note: Parsing the resource name is needed only for patterns where the main identification section does not work with the ARN of the AWS resource. When that is the case, a relation attribute has to be parsed from the AWS ARN.

- Includes references to cloud resource type. Establishes the relationship between the tag and the cloud resource type.

The pattern includes an API called “Cloud Rest Call” Post Call used to collect the following data:

- API URL used to retrieve the data — “https://tagging.” + $ldc(1).object_id + “.amazonaws.com/”
- API Body — “{"ResourceTypeFilters": ["" + $resource_type + "]}“
- Headers — “Content-Type: application/x-amz-json-1.1,X-Amz-Target:ResourceGroupsTaggingAPI_20170126.GetResources”

This data displays as shown in the following window.
2.1. Get tags

<table>
<thead>
<tr>
<th>Operation</th>
<th>Cloud REST Call</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>URL</strong></td>
<td>&quot;<a href="https://tagging.+$%7Bdc%5B1%5D.object_id%7D+amazonaws.com/">https://tagging.+${dc[1].object_id}+amazonaws.com/</a>&quot;</td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>&quot;POST&quot;</td>
</tr>
<tr>
<td><strong>Body</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Headers</strong></td>
<td>&quot;Content-Type:application/x-amz-json-1.1,X-Amz-Target:ResourceGroupsTaggingService.GetTags&quot;</td>
</tr>
<tr>
<td><strong>Formatted</strong></td>
<td>&quot;true&quot;</td>
</tr>
</tbody>
</table>

**Define Parsing**

- JSON File

**Output**

Data from Cloud Rest Call

This pattern collects the following data during horizontal discovery:

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI type (cmdb_key_value)</td>
<td></td>
</tr>
<tr>
<td>Key</td>
<td>Tag key</td>
</tr>
<tr>
<td>Value</td>
<td>Tag value</td>
</tr>
<tr>
<td>Table and field</td>
<td>Additional information</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Object_id</td>
<td>Unique resource ID that identifies the resource in the AWS console.</td>
</tr>
<tr>
<td>Class model (cmdb_key_value)</td>
<td></td>
</tr>
<tr>
<td>Extends table</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Identifier - Object_id</td>
<td>Related entries to cloud resources</td>
</tr>
<tr>
<td>Metadata rules</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

This data is displayed as shown in the following window:

![Image of Key Value window]

Data collected during horizontal discovery

AWS discovery pattern uses the following EVAL:

```javascript
var rtrn;
var name = ${cmdb_key_value[].object_id} 
name = name.substring(name.lastIndexOf("restapis/")+9,name.lastIndexOf); 
name = name.substring(0,name.indexOf("/")); 
rtrn = name;
```

Similar EVALS are available for patterns where the main identification section does not include ARN information in the main CI type table. When this information is not available, the Tag
Extension section must parse a unique ARN used to establish the relationship to the related cloud resource.

**ColdFusion discovery**

Discovery finds Adobe ColdFusion servers and the instances of ColdFusion applications running on them. Only the 2016 version of ColdFusion is supported.

**Components**

Discovery and Service Mapping use the **ColdFusion Application** and **ColdFusion Application Server** patterns to run horizontal and top-down discovery. The **ColdFusion Application Server** pattern is triggered from the ColdFusion Application Server process classifier for horizontal discovery.

**Prerequisites**

**Credentials**

Configure the following credentials, depending on what type of host machine the ColdFusion server is installed on:

- Windows credentials
- SSH credentials (for Linux machines)

**File access**

The pattern must be able to access these ColdFusion files:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>neo-runtime.xml</td>
<td>This file provides the mapping between the URL and the working directory path of the application.</td>
</tr>
<tr>
<td>application.cfc and application.cfm</td>
<td>These files provide application names.</td>
</tr>
<tr>
<td>neo-datasource.xml</td>
<td>This file provides the name of the datasource that is used in the configuration of the ColdFusion application. The datasource reference is configured in the application.cfc and application.cfm files.</td>
</tr>
</tbody>
</table>

*Note:* The datasource is used for top-down discovery only.

**Entry point**

For top-down discovery, use the full URL or IP address of the web application as the HTTP(S) endpoint. For example: [https://my-website-on-coldfusion/path](https://my-website-on-coldfusion/path) or [https:10.120.255.255:8500/path](https:10.120.255.255:8500/path)
Data collected during horizontal discovery

<table>
<thead>
<tr>
<th>Table and Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cold Fusion Server (cmdb_ci_coldfusion_server)</td>
<td>The name and version of the ColdFusion server, and the directory where it is installed.</td>
</tr>
<tr>
<td>Installation directory (installation_directory)</td>
<td>The name of the application and the directory where it is installed.</td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Version (version)</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** Cold Fusion applications (cmdb_ci_cf_application) are custom applications that this pattern does not discover.

CI relationships

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_cf_application</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_coldfusion_server</td>
</tr>
<tr>
<td>cmdb_ci_coldfusion_server</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_linux_server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmdb_ci_win_server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmdb_ci_osx_server</td>
</tr>
</tbody>
</table>

Connections discovered by Service Mapping during the top-down discovery

Service Mapping performs the top-down discovery of the Adobe ColdFusion in the context of application services. Service Mapping discovers the outgoing datasource connections from ColdFusion servers to instances of database.

**Example**

In this example, the application service map shows the results of a top-down discovery of the ColdFusion Web application service. The application service is comprised of three CIs:

- The ColdFusion application server is named `cfusion`.
- The ColdFusion application `hdStreetOracle`.
- A database named `XE`, which the ColdFusion application connects to.
**EMC Isilon discovery**

Discovery uses the EMC Isilon pattern to find components of Dell EMC Isilon. Discovery can find EMC Isilon running OneFS OS version 8.0 and later. The EMC Isilon pattern is triggered by the EMC Isilon classifier that is part of the update set you download from Store.

**Prerequisites**

**SNMP Community**

Create a SNMP Community credential for the Isilon cluster. The discovery process begins with using SNMP-based queries to classify an application as an Isilon cluster.

**EMC Isilon user**

Once the application is classified as an Isilon cluster, the discovery process uses REST queries to find information about it. Provide the EMC Isilon user with the following REST capabilities:

- "https://" + $host + ":8080/platform/3/cluster/config"
In addition to these capabilities, this EMC Isilon user must have the following privileges:

- ISI_PRIV_LOGIN_PAPI
- ISI_PRIV_AUTH
- ISI_PRIV_DEVICES
- ISI_PRIV_NETWORK
- ISI_PRIV_NFS
- ISI_PRIV_SMARTPOOLS
- ISI_PRIV_SMB

**Basic authentication**

In the Credentials module of the ServiceNow platform, create a basic authentication credential for the same EMC Isilon user.

**Upgrade the ServiceNow platform**

Upgrade the ServiceNow platform to the relevant patch: Jakarta Patch 10, Kingston Patch 8, or London Patch 1.

⚠️ **Note:** The latest patches contain metadata rules for Isilon and its components.

### Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Server (cmdb_ci_storage_server)</td>
<td>The server hosting the Isilon cluster.</td>
</tr>
<tr>
<td>short_description</td>
<td>The description configured during the Isilon server installation.</td>
</tr>
<tr>
<td>firmware_version</td>
<td>The version of the firmware installed on the Isilon server</td>
</tr>
<tr>
<td>ip_address</td>
<td>The IP address of the Isilon server.</td>
</tr>
<tr>
<td>manufacturer</td>
<td>The company that produced the device: EMC.</td>
</tr>
<tr>
<td>model_id</td>
<td>Preconfigured identification string of the device.</td>
</tr>
<tr>
<td>host_name</td>
<td>The host name of the Isilon server.</td>
</tr>
<tr>
<td>Storage Cluster (cmdb_ci_storage_cluster)</td>
<td>The name of the IP address of the cluster that Isilon servers form.</td>
</tr>
<tr>
<td>Table and Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Storage Cluster Node (cmdb_ci_storage_cluster_node)</td>
<td>The name and other attributes of the node that is part of the Isilon storage cluster.</td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>operational_status</td>
<td></td>
</tr>
<tr>
<td>Storage Node Element (cmdb_ci_storage_node_element)</td>
<td>The name and other attributes of the physical node that are hosted by the storage cluster.</td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>manufacturer</td>
<td></td>
</tr>
<tr>
<td>cpu_type</td>
<td></td>
</tr>
<tr>
<td>model_id</td>
<td></td>
</tr>
<tr>
<td>Network Adapter (cmdb_ci_network_adapter)</td>
<td>The attribute showing if the network adapter installed on the cluster node is active.</td>
</tr>
<tr>
<td>active</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>netmask</td>
<td></td>
</tr>
<tr>
<td>IP Address (cmdb_ci_ip_address)</td>
<td>The attributes of the IP address of the cluster node.</td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>netmask</td>
<td></td>
</tr>
<tr>
<td>device_id</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>model_id</td>
<td></td>
</tr>
<tr>
<td>interface</td>
<td></td>
</tr>
<tr>
<td>disk_interface</td>
<td></td>
</tr>
<tr>
<td>Storage File Share (cmdb_ci_storage_fileshare)</td>
<td>The fileshare server attributes.</td>
</tr>
<tr>
<td>path</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>fileshare_id</td>
<td></td>
</tr>
<tr>
<td>correlation_id</td>
<td></td>
</tr>
<tr>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>The attributes of the storage volume belonging to the storage cluster.</td>
</tr>
<tr>
<td>media_type</td>
<td></td>
</tr>
<tr>
<td>object_id</td>
<td></td>
</tr>
<tr>
<td>size_bytes</td>
<td></td>
</tr>
<tr>
<td>free_space_bytes</td>
<td></td>
</tr>
<tr>
<td>Storage Pool (cmdb_ci_storage_pool)</td>
<td></td>
</tr>
</tbody>
</table>
### Table and Field

<table>
<thead>
<tr>
<th>Table and Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pool_id</td>
<td>The attributes of the storage pool to which the storage cluster belongs.</td>
</tr>
<tr>
<td>free_space_bytes</td>
<td></td>
</tr>
<tr>
<td>size_bytes</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>object_id</td>
<td></td>
</tr>
<tr>
<td>Serial Number (cmdb_serial_number)</td>
<td>The attributes of the serial number of the storage node.</td>
</tr>
<tr>
<td>serial_number</td>
<td></td>
</tr>
<tr>
<td>serial_number_type</td>
<td></td>
</tr>
<tr>
<td>valid</td>
<td></td>
</tr>
</tbody>
</table>

The graphic illustrates CIs that are part of Isilon discovery.

**Note:** This Dependency Views map was simplified for clarity. Your Isilon deployments can contain many more CIs.

### CI Relationships

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship/Reference</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_storage_server</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_storage_cluster</td>
</tr>
<tr>
<td>cmdb_ci_storage_server</td>
<td>Allocated to::Allocated from</td>
<td>cmdb_ci_storage_node_element</td>
</tr>
<tr>
<td>cmdb_ci_storage_cluster_node</td>
<td>Cluster of::Cluster</td>
<td>cmdb_ci_storage_cluster</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(reference cluster)</td>
</tr>
<tr>
<td>CI</td>
<td>Relationship/Reference</td>
<td>CI</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>cmdb_ci_storage_cluster_node</td>
<td>Hosed on::Hosts</td>
<td>cmdb_ci_storage_node_element (reference server)</td>
</tr>
<tr>
<td>cmdb_ci_storage_node_element</td>
<td>Managed by::Manages</td>
<td>cmdb_ci_storage_cluster</td>
</tr>
<tr>
<td>cmdb_ci_network_adapter</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_storage_node_element (reference cmdb_ci)</td>
</tr>
<tr>
<td>cmdb_ci_network_adapter</td>
<td>Used by::Uses</td>
<td>cmdb_ci_storage_server (reference cmdb_ci)</td>
</tr>
<tr>
<td>cmdb_ci_ip_address</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_storage_node_element</td>
</tr>
<tr>
<td>cmdb_ci_ip_address</td>
<td>Used by::Uses</td>
<td>cmdb_ci_storage_server</td>
</tr>
<tr>
<td>cmdb_ci_disk</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_storage_node_element (reference computer)</td>
</tr>
<tr>
<td>cmdb_ci_storage_fileshare</td>
<td>Allocated from::Allocated to</td>
<td>cmdb_ci_disk</td>
</tr>
<tr>
<td>cmdb_ci_storage_fileshare</td>
<td>Provided by::Provides</td>
<td>cmdb_ci_storage_server</td>
</tr>
<tr>
<td>cmdb_ci_storage_fileshare</td>
<td>Allocated from::Allocated to</td>
<td>cmdb_ci_storage_volume</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Uses::Used by</td>
<td>cmdb_ci_storage_server</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_storage_cluster</td>
</tr>
<tr>
<td>cmdb_ci_storage_pool</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_storage_cluster (reference hosted_by</td>
</tr>
</tbody>
</table>

**Google Cloud Platform (GCP) asset inventory discovery**

Discovery uses the Google Cloud Platform (GCP) asset inventory pattern to find GCP resources and policies.

GCP provides container resources, such as organizations and projects, that allow other GCP resources to be grouped and organized hierarchically. The GCP Resource Manager API enables you to manage these container resources.

The pattern provides visibility for services supported by the Asset Inventory API. It collects inventory data on the deployed GCP services and updates the CMDB.

The pattern collects inventory data either for all GCP-supported resources or for a preconfigured whitelist of resources. The whitelist resides in the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table. It contains the resource types supported by GCP Cloud Asset Inventory, except for Compute Engine resources and IAM policies. For more information, see [https://cloud.google.com/resource-manager/docs/cloud-asset-inventory/overview](https://cloud.google.com/resource-manager/docs/cloud-asset-inventory/overview).

Visit the [ServiceNow Store](https://servicenow.com/store) website to view all the available apps and for information about submitting requests to the store.

**Prerequisites**

- GCP authorization to use the API [https://cloudasset.googleapis.com/v1/projects/deductive-reach-207607:exportAssets](https://cloudasset.googleapis.com/v1/projects/deductive-reach-207607:exportAssets), and one or more of the following Google IAM permissions on the specified resource parent:
  - cloudasset.assets.exportResource
  - cloudasset.assets.exportIamPolicy
- A storage bucket for storing the collected data. This bucket is used as input parameter in the pattern.
  - If a bucket must be created for this purpose, the same GCP user must have both Create and Edit access for it.
  - The Retention Policy for the storage bucket must not be active. Otherwise, the pattern does not delete the auto-generated inventory data file.

- A read-only GCP user for using the API https://www.googleapis.com/storage/v1.

  **Note:** You can use the headers on the Encryption page in your request to do the following:
  - Download an object that is encrypted by a customer-supplied encryption key.
  - Get object metadata with content hashes.

- To collect inventory data for GCP-supported resources, in Now Platform navigate to **Cloud Inventory Resource Whitelist** and clear all GCP table records.

  ![Cloud Inventory Resource Whitelist](image)

Cloud Inventory Resource Whitelist

- Fine-tune the GCP resource discovery using the Resource Whitelist.
  If your deployment has custom patterns for GCP discovery, ensure that you do not discover GCP resources twice.

  1. Ensure that the application scope is Discovery and Service Mapping Patterns:
     a. Navigate to **Settings > Developer**.
     b. Select **Discovery and Service Mapping Patterns** from the **Application** list.

  2. Navigate to **System Definitions > Tables**.

  3. Open the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table.
4. Under Related Links, click Show List.

5. Select resource types for which you have custom patterns, and select Delete from the Actions on selected rows list.

The Cloud Inventory Resource Whitelist is predefined with common services. You can expand the whitelist with additional resource types that you want the pattern to discover. The names of these resource types must conform to the appropriate vendor naming conventions.

**Note:** When you modify the out-of-the-box whitelist, it is no longer updated automatically in application updates. You need to maintain the customized list on your own.

a. Open the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table.

b. Click New.

c. Fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Vendor</td>
<td>The vendor of the resource type: GCP.</td>
</tr>
<tr>
<td>Resource Type</td>
<td>The AWS resource type value.</td>
</tr>
<tr>
<td>Application</td>
<td>The application scope: Discovery and Service Mapping Patterns.</td>
</tr>
</tbody>
</table>

The changes are applied the next time you run the pattern.

- Modify the ServiceNow identification rules. Be sure to switch to the Cloud Management Platform application scope before you begin the following procedure.
  1. Navigate to CI Class Manager.
  2. Navigate to and select Hierarchy.
  5. Delete and add the appropriate identification rule.

     The new identification rule should include only the Object ID attribute.
Schedule for all serverless discovery and pass the path of the storage bucket as a pattern variable. For more information, see Create a schedule for standard serverless Discovery.

Data collected by Discovery during horizontal discovery

This pattern discovers data that provides visibility for all GCP services in your organization. The discovered data includes the following tables and fields.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI (cmdb_ci_cmp_resource)</td>
<td>The ID of the item. The item is accessed with this URL.</td>
</tr>
<tr>
<td>object_id</td>
<td>The name of the resource.</td>
</tr>
<tr>
<td>name</td>
<td>The asset resource type, according to the data in the JSON file.</td>
</tr>
</tbody>
</table>

The Dependency Views map shows the discovered CIs in your organization and the relationships between them. Here, the only meaningful relationship between the CIs is the one that helps Discovery identify them.

Each GCP Inventory CI is related either to a Logical Datacenter (LDC) CI or to a Cloud Service Account CI. In this example, the Inventory CI is related to a Cloud Service Account CI.
CI relationships

These relationships are created to support GCP asset inventory discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>For Global Resources:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Main CI</td>
<td>Contained by::Contains</td>
<td>Cloud Service Accounts</td>
</tr>
<tr>
<td>(cmdb_ci_cmp_resource)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For Regional Resources:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Main CI</td>
<td>HostedOn::Hosts</td>
<td>Logical Datacenter (LDC)</td>
</tr>
<tr>
<td>(cmdb_ci_cmp_resource)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Google Cloud Platform discovery

ServiceNow® Discovery finds Google Cloud Platform (GCP) GCP API v1 components using the Google Cloud Platform pattern. The Google Cloud Platform pattern is a ServiceNow Store app.

Get the app on the ServiceNow Store

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.
Prerequisites
For details on system requirements, view the application listing on the ServiceNow Store website.

Cloud Management
Google Cloud Platform patterns work with Cloud Management. The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt). Use your current Cloud Management instance or request a subscription from your sales representative.

MID Server requirements
Use a dedicated MID Server for cloud discovery. The MID Server must have 16 Cores, 8 GB with 4 GB Java Memory Model (JVM) memory allocation. Deploy and connect the dedicated MID Server with internet access.

User role settings
Set the user permission for the Google Cloud Platform member to Viewer:
1. In the Google Cloud Platform console, navigate to AIM.
2. Select the relevant member from the list and click the Edit icon.
3. In the Edit permissions window, select Viewer from the Role list.
4. Click Save.

Credentials
Configure the following credentials:
1. Navigate to Discovery > Credentials.
2. Click New.
3. Click Google API Credentials, fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Credential name. An example is gcp-cred</td>
<td></td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this credential for discovery. Select this check box to enable discovery.</td>
<td></td>
</tr>
<tr>
<td>Applies to</td>
<td>MID Server to which you must apply the credentials. Select All MID servers.</td>
<td></td>
</tr>
<tr>
<td>Order</td>
<td>Order in which the platform tries this credential as it attempts to log on to devices. A smaller number indicates that the credential appears higher in the list. Establish the credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number, or none, the instance tries the credentials in a random order. The default value is 100.</td>
<td></td>
</tr>
<tr>
<td>EMail</td>
<td>GCP client email.</td>
<td></td>
</tr>
</tbody>
</table>
Create a Google Cloud Platform service account

Use the credentials defined in the preceding procedure for the service account.

1. Navigate to Cloud Management > Service Accounts.
2. Click New, fill in the form, and then click Submit.

Service account form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secret Key</td>
<td>Secret key for these credentials.</td>
</tr>
<tr>
<td>Credential alias</td>
<td>Credential alias that is used by the credentials to authorize actions.</td>
</tr>
</tbody>
</table>

Create a discovery schedule

1. Navigate to Discovery > Discovery Schedules.
2. Click New and then fill in the form.

Discovery Schedule form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the Discovery schedule. For example, gcp</td>
</tr>
<tr>
<td>Discover</td>
<td>Discover type. For the London release only, select Serverless.</td>
</tr>
<tr>
<td>MID Server</td>
<td>Name of the MID Server. For example, Discovery_Server</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this schedule for discovery. Select this check box to enable discovery.</td>
</tr>
</tbody>
</table>

3. Right-click the header of the Discovery Schedule form and select Save.
4. For the London release only, click the **Serverless Execution Patterns** tab, click **New**, and then fill in the

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for this Serverless Execution Pattern. For example, gcp1</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Pattern</td>
<td>Name of the GCP pattern to run. Select the pattern to run.</td>
</tr>
<tr>
<td>Proxy Host</td>
<td>Fully qualified domain name of the machine on which you are installing the proxy server. Specify <strong>Global</strong>.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this schedule for discovery. Select this check box to enable discovery.</td>
</tr>
</tbody>
</table>

The following parameters are required to run specific patterns:

**Discovery schedule configuration for specific patterns**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cloud_account_id</td>
<td>Name of the GCP project.</td>
</tr>
<tr>
<td>cloud_cred_id</td>
<td>Sysid of the credentials that you created for the service account.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>cloud_datacenter_type</td>
<td>CI’s name that holds the datacenters for GCP. Specify cmdb_ci_google_datacenter</td>
</tr>
</tbody>
</table>

**Discovery schedule configuration for full Google Cloud Platform discovery**

1. Navigate to **Cloud Management** > **Service Accounts**.
2. Click the required service account.
3. Click **Create Discovery Schedule**.
4. Create and define the Discovery Schedule as described in the product documentation.
   This discovery schedule runs all GCP patterns.

**Credentials for API elements**

Provide a read-only user with permission to use the following API elements:

- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/regions
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/subnetworks
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/global/firewalls
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/zones
   (For HTTP Load Balancer)
  (For TCP/UDP Load Balancer)
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/backendServices
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/instanceGroups
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/instanceGroupManagers
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/zones/<ZONE>/instanceGroups/<instance group>/listInstances
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/regions/<REGION>/instanceGroups/<instance group>/listInstances
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/instances
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/machineTypes
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/global/instanceTemplates
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/disks
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/diskTypes
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/global/healthChecks
- https://www.googleapis.com/compute/<api_version>/projects/<project_id>/aggregated/addresses

**GCP flow - horizontal discovery**

Where discovery is performed using a serverless schedule, the GCP patterns that were selected run and discover GCP resources.

Where discovery is performed using a service account, a full GCP discovery schedule is run, discovering all supported GCP resources.

**Methods to retrieve data**

Cloud REST – Add Response To Context

The operation to run a REST call against supported clouds. This operation supports pagination by automatically concatenating pages returned by the Cloud provider, Google, into one response, and saves the outcome as a list for fast parsing.

Use **Body** and **Headers** when requested by Google. Credentials are automatically inserted into the required field of the request form.

You can use:

- GCP Compute Pillar – Class Model – Service Account
- GCP Compute Pillar – Class Model – Google Datacenter
- GCP Compute Pillar – Class Model – Availability Zone
- GCP Network Pillar – Class Model – VM instance
- GCP Compute Pillar – Class Model – Storage Volume
- GCP Compute Pillar – Class Model – Image
- GCP Compute Pillar – Class Model – instance template
- GCP Compute Pillar – Class Model – NIC

(Optional) Use GCP labels

You can use GCP labels (set up as key-value pairs) to find and group resources, such as the following:

- Virtual machine instances
- Forwarding rules (Alpha)
- Images
- Persistent disks
- Persistent disk snapshots
- Static external IP addresses (Alpha)
- VPN tunnels (Alpha)

**Note:** Resources listed as Alpha are not currently supported on either the GCloud or the GCP Console.

**Limitations**

The GCP Load Balancer pattern works with these load balancer models:

- HTTP/s Load Balancer
- TCP/UDP Load Balancer

**GCP Load Balancer models**

**GCP Load Balancer – HTTP Model**

- In the HTTP LB, there are load balancer services of the frontend and backend type.
- In the HTTP LB, the health service has a relationship to the Load Balancer service (only Backend Service).
- The load balancer does not have a direct relationship to the Load balancer pool.
- Pools are groups of instances.

The following image depicts the HTTP Load Balancer
The TCP/UDP LB has the load balancer service of the front-end type only.
- In the TCP/UDP LB, the health service has a relationship to the Load Balancer pools.
- In case the TCP/UDP LB does not have instance groups (pools), a virtual pool is created with a name that includes the name of the LB + “-Pool”.
- Pools are groups of instances.

The following image depicts the TCP/UDP Load Balancer
About the event-driven GCP load balancer

Event-driven discovery is triggered only when Cloud Load Balancing receives the optional input_object_id. Note the following points:

- **For Cloud Load Balancing (HTTP(s))**: The event-driven pattern is triggered when an event is received, when a new HTTP type load balancer (url_map) is added, or when a new backend service is added for a specific load balancer.

- **For Cloud Load Balancing (TCP/UDP)**: The event-driven pattern is triggered when an event is received, when a new TCP/UDP type load balancer (target_pool) is added, or when an instance is added or removed from an instance_group (pool) belonging to a TCP/UDP load balancer.
Create a filter to use for event-driven discovery, as shown below. The filter is used in the REST API calls to retrieve the GCP data.

```
Operation: Set Parameter Value

Meet: Any

If Condition Is: true

Do:

Value: "?filter=id%3D%20" + $input_object_id

Name: $filter
```

- **For Cloud Load Balancing (HTTP):** `https://www.googleapis.com/compute/" + $API_version + "/projects/" + $service_account[1].account_id + "/global/urlMaps" + $filter

### Data collected by Discovery during horizontal discovery

Data is collected using these patterns:

- Google Cloud Platform Discovery Compute Pillar.
- Google Cloud Platform Discovery External IP Pillar.
- Google Cloud Platform Discovery Load Balancer.
- Google Cloud Platform Discovery Labels.
- Google Cloud Platform Discovery Network Pillar.
- Google Cloud Platform Discovery Storage Pillar.

### Resources discovered by the Google Cloud Platform Discovery Compute Pillar

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_service_account</td>
<td></td>
</tr>
<tr>
<td>account_id</td>
<td>Name of the project that is used for the discovery.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>object_id</td>
<td>Field with the same value as the account_id. Used by the IRE identification rules.</td>
</tr>
<tr>
<td>datacenter_type</td>
<td>Name of the CI that holds the datacenters for Google Cloud Platform.</td>
</tr>
<tr>
<td>cmdb_ci_google_datacenter</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Datacenter or region name.</td>
</tr>
<tr>
<td>region</td>
<td>Field with the same value as the name field.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Datacenter or region description that you provide for easy identification.</td>
</tr>
<tr>
<td>status</td>
<td>Datacenter or region status. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· Up</td>
</tr>
<tr>
<td></td>
<td>· Down</td>
</tr>
<tr>
<td>state</td>
<td>Field with the same value as the status field.</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the availability zone (AZ).</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Availability zone description that you provide.</td>
</tr>
<tr>
<td>cmdb_ci_vm_instance</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the Virtual Machine (VM).</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>state</td>
<td>VM state. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· On</td>
</tr>
<tr>
<td></td>
<td>· Running</td>
</tr>
<tr>
<td></td>
<td>· Off</td>
</tr>
<tr>
<td></td>
<td>· Terminated</td>
</tr>
<tr>
<td></td>
<td>· Paused</td>
</tr>
<tr>
<td>Disks and NICs</td>
<td>Number of disks and Network Interface Controllers (NICs) assigned to this VM.</td>
</tr>
<tr>
<td>CPUs and memory</td>
<td>Number of CPU cores and memory size assigned to this VM.</td>
</tr>
<tr>
<td>cmdb_ci_instance_template</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the instance template.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Description of the instance template.</td>
</tr>
<tr>
<td>source_image</td>
<td>Source image of the instance template.</td>
</tr>
</tbody>
</table>
### Related CI: cmdb_ci_nic

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the NIC.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, made up of a concatenation of vm_id + &quot;:&quot; + nic_name.</td>
</tr>
<tr>
<td>ip_address</td>
<td>NIC primary internal IP address.</td>
</tr>
<tr>
<td>state</td>
<td>Field with the same value as the GCP state field.</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_storage_volume

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the disk.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>size</td>
<td>Disk size in GB.</td>
</tr>
<tr>
<td>short_description</td>
<td>Description that you provide for easy identification.</td>
</tr>
<tr>
<td>state</td>
<td>State of the Disk. The state is always “available”, since it was returned on discovery.</td>
</tr>
<tr>
<td>status</td>
<td>Installed.</td>
</tr>
<tr>
<td>storage_type</td>
<td>Disk type. For example, pd-standard or local SSD.</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_os_template

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the image.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>image_type</td>
<td>Image type.</td>
</tr>
<tr>
<td>image_source</td>
<td>Source of the image, such as, source disk/snapshot + &quot;/&quot; + source disk/snapshot ID.</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_compute_template

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the image.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>vCPUs</td>
<td>Number of CPU cores assigned to this template.</td>
</tr>
<tr>
<td>memory_mb</td>
<td>Memory size assigned to this template.</td>
</tr>
<tr>
<td>local_storage_gb</td>
<td>Storage assigned to this template.</td>
</tr>
</tbody>
</table>

### Resources discovered by the Google Cloud Platform Discovery External IP Pillar

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_service_account</td>
<td>Name of the project that is used for the discovery.</td>
</tr>
<tr>
<td>account_id</td>
<td>Name of the project that is used for the discovery. Used by the Identification and Reconciliation Engine (IRE) identification rules.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>datacenter_type</td>
<td>Name of the CI that holds the datacenters for Google Cloud Platform</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Datacenter or region name.</td>
</tr>
<tr>
<td>region</td>
<td>Field with the same value as the name field.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Datacenter or region description that you provide for easy identification.</td>
</tr>
<tr>
<td>status</td>
<td>Datacenter or region status. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· Up</td>
</tr>
<tr>
<td></td>
<td>· Down</td>
</tr>
<tr>
<td>state</td>
<td>Field with the same value as the status field.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_public_ipaddress</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the external IP.</td>
</tr>
<tr>
<td>public_ip_address</td>
<td>Address of the public IP.</td>
</tr>
<tr>
<td>state</td>
<td>State of the external IP Address. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· In Use</td>
</tr>
<tr>
<td></td>
<td>· Available</td>
</tr>
<tr>
<td></td>
<td>Use “In Use” if the external IP is in use or “Available” if reserved</td>
</tr>
<tr>
<td></td>
<td>for later use.</td>
</tr>
</tbody>
</table>

**Resources discovered by the Google Cloud Platform Discovery Load Balancer**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_service_account</td>
<td></td>
</tr>
<tr>
<td>account_id</td>
<td>Name of the project that is used for the discovery.</td>
</tr>
<tr>
<td>object_id</td>
<td>Field with the same value as the account_id. Used by the Identification and Reconciliation Engine (IRE) identification rules.</td>
</tr>
<tr>
<td>datacenter_type</td>
<td>Name of the CI that holds the datacenters for Google Cloud Platform</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Datacenter or region name.</td>
</tr>
<tr>
<td>region</td>
<td>Region field that has the same value as the name field.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>short_description</td>
<td>Datacenter or region description that you provide for easy identification.</td>
</tr>
<tr>
<td>status</td>
<td>Datacenter or region status. One of the following:</td>
</tr>
<tr>
<td></td>
<td>• Up</td>
</tr>
<tr>
<td></td>
<td>• Down</td>
</tr>
<tr>
<td>state</td>
<td>State field that has the same value as the status field.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td>Name of the availability zone (AZ).</td>
</tr>
<tr>
<td>name</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Availability zone description that you provide.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_load_balancer</td>
<td>Name of the load_balancer.</td>
</tr>
<tr>
<td>name</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_service</td>
<td>Name of the load balancer service.</td>
</tr>
<tr>
<td>name</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>service_type</td>
<td>Service type. One of:</td>
</tr>
<tr>
<td></td>
<td>• Frontend (HTTP/TCP/UDP LB)</td>
</tr>
<tr>
<td></td>
<td>• Backend (HTTP LB only).</td>
</tr>
<tr>
<td>port</td>
<td>Listeners protocol port number (Backend).</td>
</tr>
<tr>
<td>front_end_port</td>
<td>Listeners protocol port number (Frontend).</td>
</tr>
<tr>
<td>listener_protocol</td>
<td>Listeners protocol. One of</td>
</tr>
<tr>
<td></td>
<td>• TCP</td>
</tr>
<tr>
<td></td>
<td>• HTTP</td>
</tr>
<tr>
<td></td>
<td>• HTTPS</td>
</tr>
<tr>
<td>pool</td>
<td>Pool used by the LB service.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool</td>
<td>Pool name. For TCP/UDP LB without pools, the name containing the LB name -Pool.</td>
</tr>
<tr>
<td>name</td>
<td>The unique identifier of the LB Pool. For TCP/UDP LB without pools, the name containing the LB name -Pool.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool_member</td>
<td>Name of the pool member.</td>
</tr>
<tr>
<td>name</td>
<td>Unique identifier of the Pool Member, created from the ID of the instance object_id.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_health_service</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the health service.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the Health Service Check.</td>
</tr>
<tr>
<td>monitor_type</td>
<td>Type of health check. One of:</td>
</tr>
<tr>
<td></td>
<td>- TCP</td>
</tr>
<tr>
<td></td>
<td>- HTTP</td>
</tr>
<tr>
<td></td>
<td>- HTTPS</td>
</tr>
<tr>
<td>timeout_sec</td>
<td>Timeout in seconds to wait for health checks response.</td>
</tr>
<tr>
<td>check_interval_sec</td>
<td>Interval in seconds to perform health check.</td>
</tr>
<tr>
<td>healthy_threshold</td>
<td>Number of sequential successful probe results for a VM instance to be considered healthy.</td>
</tr>
<tr>
<td>unhealthy_threshold</td>
<td>Number of sequential failed probe results for a VM instance to be considered unhealthy.</td>
</tr>
<tr>
<td>request_path</td>
<td>URL path to which GCP sends health check probe requests.</td>
</tr>
<tr>
<td></td>
<td>If omitted, GCP sends probe requests to the root path.</td>
</tr>
<tr>
<td>port</td>
<td>Protocol and port to determine how GCP health check systems contact the VMs.</td>
</tr>
</tbody>
</table>

**Resources discovered by Google Cloud Platform Discovery using labels**

When Google Cloud Platform Discovery uses labels during discovery, the following data is collected from the Key Value form:

- Key
- Value
- Configuration item
Key Value form

This data is identified in the Related Entry form for the following CMDB tables:

- Virtual Machine Instance table (cmdb_ci_vm_instance) as shown below
- Image table (cmdb_ci_os_template)
- Storage Volume table (cmdb_ci_storage_volume)
- Storage Snapshot table (cmdb_ci_storage_vol_snapshot as inherited from the parent)
### Resources discovered by the Google Cloud Platform Discovery Network Pillar

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_service_account</td>
<td>Name of the project that is used for the discovery.</td>
</tr>
<tr>
<td>account_id</td>
<td>Field with the same value as the account_id. Used by the IRE identification rules.</td>
</tr>
<tr>
<td>object_id</td>
<td></td>
</tr>
<tr>
<td>datacenter_type</td>
<td>Name of the CI that holds the datacenters for Google Cloud Platform cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Datacenter or region name.</td>
</tr>
<tr>
<td>region</td>
<td>Field with the same value as the name field.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Datacenter or region description that you provide for easy identification.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| status | Datacenter or region status. One of the following:  
- Up  
- Down |
| state | Field with the same value as the status field. |
| name | Name of the availability zone. |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |
| short_description | Availability zone that you provide. |

**Related CI:** cmdb_ci_availability_zone

| name | Name of the Virtual Private Cloud (VPC) network. |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |
| default_gateway | Gateway address for default routing out of the network. |
| state | State of the network. The state is always “available”, since it was returned on discovery. |

**Related CI:** cmdb_ci_network

| name | Name of the instance template. |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |
| cidr | CIDR representation of the subnet. For example, 10.0.0.0/24 |
| subnet_mask | Dotted representation of the subnet mask. For example, 255.255.255.0 |
| gateway | Gateway address for default routing out of the network. |
| broadcast_address | Broadcast address of the subnet. |
| available_ip_count | Number of IPs that are available in the subnet. This amount does not include network and broadcast addresses. |

**Related CI:** cmdb_ci_cloud_subnet

| name | Name of the access control list (ACL). |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |
| short_description | ACL description that you provide. |

**Related CI:** cmdb_ci_network_acl

<p>| name | ACL rule name. This name is the same as the name of the cmdb_ci_network_acl. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| allow_deny            | ACL rule behavior. One of:  
• Allowed  
• Denied                                                                                                                                  |
| is_outbound           | Rule for outbound traffic. Use one of the following:  
• True if the rule is for outbound traffic  
• False if the rule is for inbound traffic                                                                                      |
| source_ranges         | JSON blob service that contains sources that are Classless Inter-Domain Routing (CIDR) representations that the rule either allows or blocks the traffic from the service.                                           |
| destination_ranges    | JSON blob service that contains destinations that are CIDR representations that the rule either allows or blocks the traffic from the service.                                                             |
| target_tags           | JSON blob service that represents a list of tags that controls which instances the firewall rule applies to.                                                                                                    |
| allowed_deny_traffic  | JSON blob service that contains the rules that are either allowed or denied by this ACL rule.                                                                                                               |

**Related CI:** cmdb_ci_os_template

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the image.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>image_type</td>
<td>Image type.</td>
</tr>
<tr>
<td>image_source</td>
<td>Source of the image, such as, source disk/snapshot + &quot;/&quot; + source disk/snapshot ID.</td>
</tr>
</tbody>
</table>

**Related CI:** cmdb_ci_compute_template

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the image.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier, allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>vCPUs</td>
<td>Number of CPU cores assigned to this template.</td>
</tr>
<tr>
<td>memory_mb</td>
<td>Memory size assigned to this template.</td>
</tr>
<tr>
<td>local_storage_gb</td>
<td>Storage assigned to this template.</td>
</tr>
</tbody>
</table>

**Resources discovered by the Google Cloud Platform Discovery Storage Pillar**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_service_account</td>
<td>Name of the project that is used for the discovery.</td>
</tr>
<tr>
<td>account_id</td>
<td>Field with the same value as the account_id. Used by the IRE identification rules.</td>
</tr>
<tr>
<td>datacenter_type</td>
<td>Name of the CI that holds the datacenters for Google Cloud Platform.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Related CI: <code>cmdb_ci_google_datacenter</code></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Datacenter or region name.</td>
</tr>
<tr>
<td>region</td>
<td>Field with the same value as the name field.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier allocated by Google Cloud for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Datacenter or region description that you provide for easy identification.</td>
</tr>
</tbody>
</table>
| status | Datacenter or region status. One of the following:  
- Up  
- Down |  |
<p>| state | Field with the same value as the status field. |  |
| Related CI: <code>cmdb_ci_availability_zone</code> | |  |
| name | Name of the availability zone. |  |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |  |
| short_description | Availability zone that you provide. |  |
| Related CI: <code>cmdb_ci_storage_volume</code> | |  |
| name | Name of the storage disk. |  |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |  |
| size | Disk size in GB. |  |
| state | State of the network. The state is always &quot;available&quot;, since it was returned on discovery. |  |
| status | Disk status: Installed |  |
| storage_type | Disk type. For example, pd-standard or local-ssd. |  |
| Related CI: <code>cmdb_ci_disk_type</code> | |  |
| name | Name of the disk, in the format: <code>&lt;disk type name&gt;@&lt;zone name&gt;</code> |  |
| object_id | Disk type is unique per zone. Format used: <code>&lt;zone&gt;:&lt;disk type name&gt;</code> |  |
| short_description | Disk type description. For example, Local-SSD. |  |
| valid_disk_size | Disk sizes that are valid for this disk type. |  |
| default_disk_size_gb | Default disk size (in GB) for this disk type. |  |
| Related CI: <code>cmdb_ci_storage_vol_snapshot</code> | |  |
| name | Name of the snapshot. |  |
| object_id | Unique identifier, allocated by Google Cloud for this resource. |  |
| short_description | ACL description that you provide. |  |
| state | Field with the same value as the GCP state. |  |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volume_name</td>
<td>Name of the source disk: cmdb_ci_storage_volume</td>
</tr>
<tr>
<td>capacity</td>
<td>Number of bytes stored in the snapshot.</td>
</tr>
</tbody>
</table>

**CI Relationships**

These relationships are created to support Google Cloud Platform discovery.

**Relationships for resources that Google Cloud Platform Compute Pillar discovers**

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_google_datacenter</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td>Contains::Contained By</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_vm_instance</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_vm_instance</td>
<td>Contains::Contained By</td>
<td>cmdb_ci_availability_zone</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_instance_template</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_instance_template</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td></td>
<td>Uses::Used by</td>
<td>cmdb_ci_compute_template</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cmdb_ci_os_template</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_nic</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_nic</td>
<td>implement endpoint with</td>
<td>cmdb_ci_endpoint_vnic</td>
</tr>
<tr>
<td></td>
<td>implement endpoint from</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_storage_volume</td>
<td>Contains::Contained</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td></td>
<td>implement endpoint to</td>
<td>cmdb_ci_endpoint_block</td>
</tr>
<tr>
<td></td>
<td>implement endpoint from</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_os_template</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_compute_template</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_compute_template</td>
<td>provisioned from</td>
<td>cmdb_ci_compute_template</td>
</tr>
</tbody>
</table>

**Relationships for resources that Google Cloud Platform IP Pillar discovers**
### Relationships for resources that Google Cloud Platform Load Balancer discovers

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_google_datacenter</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained By</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_public_ipaddress</td>
<td></td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_cloud_public_ipaddress</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_cloud_service_account</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_google_datacenter</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td></td>
<td>cmdb_ci_cloud_public_ipaddress</td>
</tr>
<tr>
<td>cmdb_ci_cloud_public_ipaddress</td>
<td></td>
<td>cmdb_ci_google_datacenter</td>
</tr>
</tbody>
</table>

### Relationships for resources that Google Cloud Platform Network Pillar discovers

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_google_datacenter</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained By</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_service_account</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_cloud_service_account</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool</td>
<td></td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool_member</td>
<td></td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_service</td>
<td></td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_service_For_HTTP_LB</td>
<td></td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Uses::Used by</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_load_balancer_For_TCP_UDP_LB</td>
<td></td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_cloud_load_balancer</td>
<td>Uses::Used by</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_health_service_For_HTTP_LB</td>
<td></td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_health_service</td>
<td>Uses::Used by</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_health_service_For_TCP_UDP_LB</td>
<td></td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_health_service</td>
<td>Uses::Used by</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. 
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Relationships for resources that Google Cloud Platform Storage Pillar discovers

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_google_datacenter</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_google_datacenter</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td></td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_storage_volume</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Contains::Contained</td>
<td>cmdb_ci_google_datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_disk_type</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_disk_type</td>
<td></td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_storage_vol_snapshot</td>
<td></td>
<td>cmdb_ci_storage_volume</td>
</tr>
<tr>
<td>cmdb_ci_storage_vol_snapshot</td>
<td>Host::Hosted On</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td></td>
<td>Provisioned From:: Provisioned with</td>
<td>cmdb_ci_storage_volume</td>
</tr>
<tr>
<td></td>
<td>This relationship means that the specific disk was created by this specific snapshot.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Replicates to::Replicated by</td>
<td>cmdb_ci_storage_volume</td>
</tr>
<tr>
<td></td>
<td>This relationship means that this specific snapshot was created from this specific disk.</td>
<td></td>
</tr>
</tbody>
</table>

### IBM Cloud Platform discovery

Discovery finds IBM Cloud Platform components (Softlayer API v3 and v3.1 and Bluemix API v2) using the IBM Cloud Platform patterns. You can use the patterns on the Now Platform using London Patch 7, Madrid Patch 2, or later releases.
Prerequisites

Cloud Management Platform plugin
IBM Cloud Platform patterns work with the Cloud Management Platform. The Cloud Management Platform application is available as a separate subscription and requires the Cloud Management Platform plugin (com.snc.cloud.mgmt). Use your current Cloud Management Platform instance or request a subscription from your sales representative.

MID Server
Use a dedicated MID Server for cloud discovery. The MID Server must have 16 Cores, 8 GB with 4 GB Java Memory Model (JVM) memory allocation. Deploy and connect the dedicated MID Server with internet access.

Credentials
The cloud resources are in an IBM cloud. Create the following credentials that can access your IBM account.
1. Navigate to Discovery > Credentials.
2. Click New.
3. Click IBM Credentials and then fill in the form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Credential name. An example is ibm-cred</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which the platform tries this credential as it attempts to log on to devices. A smaller number indicates that the credential appears higher in the list. Establish the credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number, or none, the instance tries the credentials in a random order. The default value is 100.</td>
</tr>
<tr>
<td>Softlayer API User</td>
<td>Username for the Softlayer API. Use the Softlayer API Username defined in the Softlayer console.</td>
</tr>
<tr>
<td>Softlayer API Key</td>
<td>Key for the Softlayer API. Use the Softlayer API Key that is defined in the Softlayer console.</td>
</tr>
<tr>
<td>Bluemix API Key</td>
<td>Key for the Bluemix API. Use the Bluemix API Key defined in the IBM Cloud console.</td>
</tr>
</tbody>
</table>

4. Click Submit.

Create an IBM cloud service account
Use the credentials defined in the preceding procedure to create the IBM cloud service account.
1. Navigate to Cloud Management > Service Accounts.
2. Click New.
3. On the form, fill in the fields:

Cloud Service Account form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Credential name. An example is ibm-snow</td>
</tr>
<tr>
<td>Account Id</td>
<td>IBM account ID name. Enter the IBM Account ID.</td>
</tr>
<tr>
<td>Discovery credentials</td>
<td>Discovery credentials that you defined earlier.</td>
</tr>
<tr>
<td>Datacenter Type</td>
<td>Datacenter type. As an example, IBM Datacenter [cmdb_ci_ibm_datacenter]</td>
</tr>
</tbody>
</table>

4. Click Submit.

Create a serverless Discovery schedule

For the London release, configure a serverless Discovery.

1. On the Now Platform, navigate to Discovery > Discovery Schedules.
2. Click New.
3. On the form, fill in the fields:

Discovery Schedule form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for this Discovery schedule. As an example, Serverless Pattern</td>
</tr>
<tr>
<td>Discover</td>
<td>Discover type. Select Serverless</td>
</tr>
<tr>
<td>MID Server</td>
<td>Name of the MID Server. As an example, Discovery_Server</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this schedule for Discovery. Select this check box to enable Discovery.</td>
</tr>
<tr>
<td>Run</td>
<td>Specifies how frequently to run the Discovery schedule.</td>
</tr>
</tbody>
</table>

4. Right-click the header of the Discovery Schedule form and select Save.
5. Click the **Serverless Execution Patterns** tab.
6. Click **New**.
7. On the form, fill in the fields:

**Serverless Execution Pattern form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for this Serverless Execution Pattern. As an example, IBM1</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Pattern</td>
<td>Name of the pattern that you want to run.</td>
</tr>
<tr>
<td>Proxy Host</td>
<td>Fully qualified domain name of the machine on which you are installing the proxy server. Specify Global.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this schedule for Discovery. Select this check box to enable Discovery.</td>
</tr>
</tbody>
</table>

8. Configure the parameters required by the IBM Cloud Platform pattern as follows:

**Discovery Pattern Launcher parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cloud_cred_id</td>
<td>The sys_id of the IBM credentials specified into the configuration.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>cloud_account_id</td>
<td>The IBM account ID.</td>
</tr>
<tr>
<td>cloud_datacenter_type</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
</tbody>
</table>

9. **Update.**

10. To start Discovery, navigate to the Serverless Discovery definition and click **Discover Now**.

**IBM flow - horizontal discovery**

- To perform a discovery using a serverless schedule, the IBM Cloud Platform patterns that were selected run and discover IBM Cloud Platform resources.
- To perform a discovery using a service account, a full IBM Cloud Platform discovery schedule runs, discovering all supported IBM Cloud Platform resources.

**Cloud custom operation**

Cloud REST – Add Response to Context

Operation for Cloud REST queries that supports large-scale responses and better memory usage.

- Specify all the fields with the same values as in the Cloud REST Query custom operation, except to specify the table for the response, which is set in the **Response Variable Name** field.

- Set **Define parsing** to None.
Methods to retrieve data for the IBM account

1. Navigate to **Pattern Designer > Custom Operations**.

2. Click **Cloud REST – add response to context**.

Method: Get Call

API URLs used to retrieve the data:

- **Load Balancer**
  ```
  https://api.softlayer.com/rest<api_version>/
  SoftLayer_Network_LBaaS_LoadBalancer/getAllObjects?
  objectMask=mask[listeners[defaultPool],members,healthMonitors]
  ```

- **Network**
  ```
  https://api.softlayer.com/rest/<API_VERSION>/SoftLayer_Account/
  getNetworkVlans?
  objectMask=mask[primaryRouter.datacenter,name,id,vlanNumber,networkSpace]
  ```

- **Network**
  ```
  https://api.softlayer.com/rest/<API_VERSION>/SoftLayer_Account/
  getSubnets?objectMask=mask[datacenter]
  ```

- **Network**
  ```
  https://api.softlayer.com/rest/<API_VERSION>/SoftLayer_Account/
  getSecurityGroups?objectMask=mask[rules]
  ```

- **Organizations and Spaces**
  ```
  https://api.<region>.bluemix.net/v2/organizations
  ```

- **Organizations and Spaces**
  ```
  https://api.<region>.bluemix.net/v2/spaces
  ```

- **SSH Key**
  ```
  SoftLayer_Account/getSshKeys
  ```

- **Storage**
  ```
  https://api.softlayer.com/rest/<api_version>/SoftLayer_Account/
  getNetworkStorage
  ```

- **Virtual Server**
  ```
  https://api.softlayer.com/rest/v3.1/SoftLayer_Account/
  getSecurityGroups
  ```

- **Virtual Server**
  ```
  http://api.softlayer.com/rest/v3.1/SoftLayer_Account/getSecurityGroups?
  objectMask=mask[id,name,networkComponentBindings[networkComponentId,networkComponent
  ```

- **Virtual Server**
  ```
  https://api.softlayer.com/rest/v3.1/SoftLayer_Account/getVirtualGuests?
  objectMask=mask[hostname,domain,powerState,fullyQualifiedDomainName,notes,id,statusId,
  ```

**EVAL functions**
Use the following EVAL function to transform the Load Balancer API response into a full JSON format. The default response is formatted as "[<resource objects>]".

```javascript
var cloud_lb_response = ${cloud_lb_response};
for (var i = 0; i < cloud_lb_response.size(); i++){
    cloud_lb_response.set(i, '{"result":[' + cloud_lb_response.get(i) + ']}');
}CTX.setAttribute("cloud_lb_response", cloud_lb_response);
```

Use the following EVAL function to transform the Network API response into a full JSON format. The default response is formatted as "[<resource objects>]".

```javascript
var network_response = ${network_response};
for (var i = 0; i < network_response.size(); i++){
    network_response.set(i, '{"result":[' + network_response.get(i) + ']}');
}CTX.setAttribute("network_response", network_response);
```

Use the following EVAL function to get the IBM region from the LB availability zone (datacenter). The same function is used for other resources inside the pattern.

```javascript
var res = "";
res = IbmCloudDcToRegion.convertDctoRegion(${cloud_lb[].cloud_lb_az_short_name});
```

Use the following EVAL function to select the required algorithm in the pattern.

```javascript
var rtrn
var lb_algorithm = ${cloud_lb_pools[].lb_algorithm}.toString();
switch (lb_algorithm){
    case "ROUNDROBIN":
        lb_algorithm = "round-robin";
        break;
    case "WEIGHTED_RR":
        lb_algorithm = "round-robin";
        break;
    case "LEASTCONNECTION":
        lb_algorithm = "least-connections";
        break;
    default: break;
}
```

Use the following EVAL function to parse the unique region to the account that hosts the Organizations and the corresponding spaces.

```javascript
var tableWithoutDuplicates = "";
tableWithoutDuplicates =  
    DuplicateRemover.removeDuplicates(${account_region},["region_name"]);
CTX.setAttribute("account_region_unique", tableWithoutDuplicates);
```

Use the following EVAL function to get information for all the organizations that are assigned to the IBM cloud account, where iteration for getting the unique region is performed.

```javascript
var cloudcall = new CloudRestAPIClient();
var resourceTable = ${account_region_unique[*].region_name};
var tableSize = resourceTable.size();
var tmp_response = new Packages.java.util.ArrayList();
var all_organizations_response = new Packages.java.util.ArrayList();
```
Use the following EVAL function to get information for all the IBM cloud account assigned spaces, where the iteration for getting the unique region is performed.

```javascript
var rtrn = ''; var cloudcall = new CloudRestAPIClient(); var resultList = new Packages.java.util.ArrayList(); var rest = ''; var resourceTable = ${account_region_unique[*].region_name}; var result = new Packages.java.util.ArrayList(); var all_organizations_response = ${all_organizations_response}
for (var i = 0; i < resourceTable.size(); i++){
    var url = "https://api."+resourceTable.get(i)+".bluemix.net/v2/spaces";
    result = cloudcall.arrayListExecute(${service_account}, null, url, null, 'GET', null, 'true', CTX);
    for (var j = 0; j < result.size(); j++){
        var jsonObj = JSON.parse(result.get(j));
        ms.log("length " + jsonObj.resources.length);
        for(var k=0;k<jsonObj.resources.length;k++){
            var gui = jsonObj.resources[k].metadata.guid;
            var url = jsonObj.resources[k].metadata.url;
            var name = jsonObj.resources[k].entity.name;
            var organization_guid = jsonObj.resources[k].entity.organization_guid;
            var allow_ssh = jsonObj.resources[k].entity.allow_ssh ? "true" : "false";
            var row = new Packages.java.util.HashMap();
            Packages.com.snc.sw.log.DiscoLog.getLogger("IBM Cloud Space and Org").debug(">>>>>>>>>>>>> region alias=" + resourceTable.get(i));
            Packages.com.snc.sw.log.DiscoLog.getLogger("IBM Cloud Space and Org").debug(">>>>>>>>>>>>> region full name=" + IbmCloudDcToRegion.regionNamebyAlias(resourceTable.get(i)));
            row.put("region", IbmCloudDcToRegion.regionNamebyAlias(resourceTable.get(i)));
            row.put("space_guid", gui);
            row.put("space_url", url);
            row.put("space_name", name);
            row.put("space_organization_guid", organization_guid);
            row.put("space_allow_ssh", allow_ssh);
            row.put("rest", rest);
            resultList.add(row);
        }
    }
}
CTX.setAttribute('spaces_table', resultList);
```
Use the following EVAL function to filter all the spaces by unique region.

```javascript
var space_region = ${spaces_table[].region};

Packages.com.snc.sw.log.DiscoLog.getLogger("IBM Cloud Space and Org").debug("filtering space according to LDC. space_region=" + space_region);
var regions = ${cmdb_ci_ibm_datacenter[*].region};
Packages.com.snc.sw.log.DiscoLog.getLogger("IBM Cloud Space and Org").debug("input regions list=" + regions.toString());

var rtrn = "false";
if(regions.contains(space_region)) {
    rtrn = "true";
}

rtrn;
```

Use the following EVAL function to filter all the spaces by storage discovery.

```javascript
var storage_response = ${storage_response};
for (var i = 0; i < storage_response.size(); i++){
    storage_response.set(i, '{"result":' + storage_response.get(i) + '}');
}
CTX.setAttribute("storage_response", storage_response);
```

Use the following EVAL function to transform the API response into full JSON format, as the untouched response is formatted as "<resource objects>".

```javascript
var ssh_key = ${ssh_key_json};
for(var i = 0; i < ssh_key.size(); i++){
    ssh_key.set(i,'"result" : ' + ssh_key.get(i) + ");
}
CTX.setAttribute("ssh_key_json", ssh_key);
```

Use the following EVAL function to populate the availability zone into the storage volume table.

```javascript
var rtrn;
var resource_group = ${network_storage[].resource_name};
var regex = /.*-(\w{3}\d{2}).*/;
var az_name = regex.exec(resource_group);
if (az_name) {
    rtrn = az_name[1];
} else {
    rtrn;
}
```

Use the following EVAL function to populate the region into the storage volume table.

```javascript
var res = ""
res = IbmCloudDcToRegion.convertDctoRegion(${network_storage[].availability_zone});
```

Use the following EVAL function to transform the size of the storages to bytes from gigabytes.

```javascript
var rtrn = ""
var size = ${network_storage[].size};
```
rtrn = (parseInt(size) * 1073741824) + '';

Use the following EVAL function to transform the size of the storages to bytes from gigabytes.

```javascript
var cloudcall = new CloudRestAPIClient();
var resourceTable = ${network_storage[*].storage_object_id};
var tableSize = resourceTable.size();
var allResources = new Packages.java.util.ArrayList();
var response = new Packages.java.util.ArrayList();
for (var i = 0; i < tableSize ; i++){
    if  (!resourceTable.get(i)) continue;
    var url = "https://api.softlayer.com/rest/"+${api_version}="/SoftLayer_Network_Storage/"+resourceTable.get(i)+"/getObject?objectMask=mask[snapshots];";
    // the commented out code is to call the "Cloud Rest Call" API
    allResources = cloudcall.arrayListExecute(${service_account}, null, url, null, 'GET', null, 'true', CTX);
    for (var j = 0; j < allResources.size(); j++){
        response.add('"result":[' + allResources.get(j) + ']}');
    }
}
CTX.setAttribute("all_snapshots_response", response);
```

Use the following EVAL closure that transforms the JSON that is received from the API to a format that the JSON parser can work with. The default JSON is saved in `vm_json_array`, the new JSON is in `vm_json`.

```javascript
var vm_json = new Packages.java.util.ArrayList();
vm_json.addAll(${vm_json_array});
for (var i = 0 ; i < vm_json.size(); i++){
    vm_json.set(i, '{"result":[' + vm_json.get(i) + ']}');
}
CTX.setAttribute("vm_json", vm_json);
```

Use the following EVAL closure that returns `on` if the power status is running. The value `off` is returned if the power status is off.

```javascript
var power = ${cmdb_ci_vm_instance[].state};
var rtrn = '';
if (power == 'Running'){
    rtrn = "on";
} else {
    rtrn = "off"
}
```

Use the following EVAL closure that returns the region of the CI, based on provided availability zone name.

```javascript
var az = ${cmdb_ci_vm_instance[].avail_zone};
var rtrn = '';
rtrn = IbmCloudDcToRegion.convertDctoRegion(az);
```

Use the following EVAL closure that builds a HashMap of HashSets, Connecting VMs with their network storages. The following key is used:

- P = page
\* V = VM \\
\* N = network storage

```javascript
var rtrn = ''; 
var vmJson = $(vm_json_array);
var VMtoStorageHashMap = new Packages.java.util.HashMap();

for (var p = 0 ; p < vmJson.size(); p++){
    var page = JSON.parse(vmJson.get(p));
    for (var v in page){
        if (page[v].allowedNetworkStorage.length > 0){
            var vm_id = page[v].id + '';
            var storageSet = new Packages.java.util.HashSet();
            for (var n in page[v].allowedNetworkStorage){
                storageSet.add(page[v].allowedNetworkStorage[n].id + '');
            }
            VMtoStorageHashMap.put(vm_id,storageSet);
        }
    }
}
CTX.setAttribute("VMStorageCache", VMtoStorageHashMap);
```

**Use the following EVAL closure to remove duplicate entries from tables.**

```javascript
var tableWithoutDuplicates = '';
tableWithoutDuplicates = DuplicateRemover.removeDuplicates($(cmdb_ci_os_template),['object_id']);
CTX.setAttribute("cmdb_ci_os_template", tableWithoutDuplicates);
```

**Use the following EVAL closure that builds a list of VM IDs that are related to the current security group ID.**

```javascript
var rtrn = '';
//get the Security Group ID of the current record in the table for comparison
var current_table_id = $(security_groups[].sec_group_id);
var current_table_id_string = current_table_id + '';

// get the list of instance's ID that will be then added to Security Group table
var needed_value_id_list = $(sec_group_network_components[*].vm_instance_id);
//get the list of Security Group IDs in the Network Component table, which will then be compared one by one to the current record of the security_groups table
var compare_id_list = $(sec_group_network_components[*].sec_group_id);

var beforeReturn = '';
var separator = '';
for (var i = 0; i < compare_id_list.size(); i++) {
    if   (current_table_id_string == compare_id_list.get(i)){
        beforeReturn += separator + needed_value_id_list.get(i);
        separator = ',';
    }
}
rtrn = beforeReturn;
```
Use the following EVAL closure that builds a HashMap of HashSet between security groups and the VMs that are configured to use them.

```javascript
rtrn = ''; 
var secGroupTable = ${cmdb_ci_endpoint_comp_security};
var secToVMHashSet = new Packages.java.util.HashMap();

// this loop worst case O(N*M) M = security groups, N = VMs
for(var i=0; i<secGroupTable.size(); i++){
    var securityTableId = secGroupTable.get(i).get('object_id');
    var vmListStr = secGroupTable.get(i).get('instance_list');
    var vmList = vmListStr.split(',');
    var vmSet = new Packages.java.util.HashSet();
    for (var j in vmList){
        vmSet.add(vmList[j]);
    }
    secToVMHashSet.put(securityTableId,vmSet);
}
CTX.setAttribute("securityVMCache", secToVMHashSet);
```

Use the following EVAL closure that returns ‘true’ if the current VM is using the current security group ID.

```javascript
var rtrn = ''; 
var vm_id = ${cmdb_ci_vm_instance[].object_id};
var vmHashSet = ${securityVMCache}.get(${cmdb_ci_endpoint_comp_security[].object_id});
rtrn = vmHashSet.contains(vm_id);
```

Use the following EVAL closure that returns ‘true’ if the current VM is using the current storage volume.

```javascript
var rtrn = ''; 
var vm_id = ${cmdb_ci_vm_instance[].object_id};
var vol_id = ${cmdb_ci_endpoint_iscsi[].object_id};
var vmHashSet = ${VMStorageCache}.get(vm_id);
if (vmHashSet){
    rtrn = vmHashSet.contains(vol_id);
} else {
    rtrn = 'false'
}
```

Use the following EVAL function to transform the API response into full JSON format, as the untouched response is formatted as "<resource objects>".

```javascript
var cloud_response = ${location_groups_response}
for (var i = 0; i < cloud_response.size(); i++){
    cloud_response.set(I, '"result":[' + cloud_response.get(i) + ']')
}
CTX.setAttribute("location_groups_response", cloud_response);
```

 Parses the unique Location Groups in the “location_groups_az” table basdon the column “lg_id” and creates a new table “location_groups_az_unique” that contains the result.

```javascript
var tableWithoutDuplicates= '';
tableWithoutDuplicates = 
    DuplicateRemover.removeDuplicates(${location_groups_az},${"lg_id"});
```
Creates a list of Availability Zones per Location Group, that is stored in a column new column in the location_groups_az_unique table. The list records are divided by commas purposely, so that unique relations could be made in consecutive step.

```javascript
CTX.setAttribute("location_groups_az_unique", tableWithoutDuplicates);

var current_table_id = ${location_groups_az_unique[].lg_id};
var current_table_id_string = current_table_id + "";
var needed_value_id_list = ${location_groups_az[*].lg_az_id};
var compare_id_list = ${location_groups_az[*].lg_id};
var beforeReturn = "";
var separator = ",";
for (var i = 0; i < compare_id_list.size(); i++) {
  if (current_table_id_string == compare_id_list.get(i)) {
    beforeReturn += separator + needed_value_id_list.get(i) + separator;
  }
}
return = beforeReturn;
```

**Data collected by Discovery during horizontal discovery**

Data is collected using these patterns:
- IBM Cloud Load Balancer
- IBM Cloud Location Groups
- IBM Cloud Network
- IBM Cloud Organizations and Spaces
- IBM Cloud Resource Groups
- IBM Cloud SSH Key
- IBM Cloud Storage
- IBM Cloud Virtual Server

**IBM Cloud Load Balancer discovered resources**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main CI: cmdb_ci_cloud_load_balancer</strong></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Load balancer name.</td>
</tr>
<tr>
<td>object_id</td>
<td>Load balancer unique identifier.</td>
</tr>
<tr>
<td>fqdn</td>
<td>Load balancer address (host name).</td>
</tr>
<tr>
<td>dns_name</td>
<td>Load balancer domain server name.</td>
</tr>
<tr>
<td>short_description</td>
<td>Load balancer description that you provide for easy identification.</td>
</tr>
<tr>
<td>operational_status</td>
<td>Load balancer status, one of either:</td>
</tr>
<tr>
<td>state</td>
<td>Load balancer provisioning status.</td>
</tr>
</tbody>
</table>

Related CI: cmdb_ci_lb_service
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Pattern created name, composed of:</td>
</tr>
<tr>
<td></td>
<td>· load balancer name</td>
</tr>
<tr>
<td></td>
<td>· type of protocol</td>
</tr>
<tr>
<td></td>
<td>· resource type listener</td>
</tr>
<tr>
<td>object_id</td>
<td>Load balancer unique identifier.</td>
</tr>
<tr>
<td>port</td>
<td>Port number of the listener.</td>
</tr>
<tr>
<td>front_end_port</td>
<td>Protocol port number of the listener.</td>
</tr>
<tr>
<td>listener_protocol</td>
<td>Protocol of the listener. One of the following protocols:</td>
</tr>
<tr>
<td></td>
<td>· TCP</td>
</tr>
<tr>
<td></td>
<td>· HTTP</td>
</tr>
<tr>
<td></td>
<td>· HTTPS</td>
</tr>
<tr>
<td>operational_status</td>
<td>Provisioning status of the listener.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Pattern created name, composed of:</td>
</tr>
<tr>
<td></td>
<td>· load balancer name</td>
</tr>
<tr>
<td></td>
<td>· type of protocol</td>
</tr>
<tr>
<td></td>
<td>· port</td>
</tr>
<tr>
<td></td>
<td>· resource type “pool”</td>
</tr>
<tr>
<td>object_id</td>
<td>Load balancer pool unique identifier.</td>
</tr>
<tr>
<td>port</td>
<td>Protocol port of the backend.</td>
</tr>
<tr>
<td>pool_status</td>
<td>Load balancer pool provisioning status.</td>
</tr>
<tr>
<td>load_balancing_method</td>
<td>Load balancing method. One of the following algorithms:</td>
</tr>
<tr>
<td></td>
<td>· “ROUNDROBIN”</td>
</tr>
<tr>
<td></td>
<td>· “WEIGHTED_RR”</td>
</tr>
<tr>
<td></td>
<td>· “LEASTCONNECTION”</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool_member</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Pattern created name, composed of:</td>
</tr>
<tr>
<td></td>
<td>· load balancer name</td>
</tr>
<tr>
<td></td>
<td>· type of protocol</td>
</tr>
<tr>
<td></td>
<td>· IP address of the Pool Member</td>
</tr>
<tr>
<td>object_id</td>
<td>Load balancer pool member unique identifier, composed of:</td>
</tr>
<tr>
<td></td>
<td>· Member ID</td>
</tr>
<tr>
<td></td>
<td>· Listener ID</td>
</tr>
<tr>
<td>ip_address</td>
<td>IP address of a load balancer member.</td>
</tr>
<tr>
<td>server_id</td>
<td>ID of the member in IBM cloud.</td>
</tr>
<tr>
<td>load_balancer</td>
<td>Name of the related Load Balancer.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_health_service</td>
<td></td>
</tr>
<tr>
<td>--------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Field</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>name</td>
<td>Pattern created name, composed of:</td>
</tr>
<tr>
<td></td>
<td>· load balancer name</td>
</tr>
<tr>
<td></td>
<td>· type of protocol</td>
</tr>
<tr>
<td></td>
<td>· resource type: <strong>Health Check</strong></td>
</tr>
<tr>
<td></td>
<td>· resource ID</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the Health Service Check.</td>
</tr>
<tr>
<td>monitor_type</td>
<td>Health check type. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· TCP</td>
</tr>
<tr>
<td></td>
<td>· HTTP</td>
</tr>
<tr>
<td></td>
<td>· HTTPS</td>
</tr>
<tr>
<td>timeout_sec</td>
<td>Timeout, in seconds, to wait for health check responses.</td>
</tr>
<tr>
<td>check_interval_sec</td>
<td>Interval, in seconds, to perform health check.</td>
</tr>
</tbody>
</table>

**IBM Cloud Location Groups discovered resources**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_location_group</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>A descriptive name used to identify a Location Group.</td>
</tr>
<tr>
<td>object_id</td>
<td>The ID of the Location Group record.</td>
</tr>
<tr>
<td>locationGroupType Name</td>
<td>The name of the Location Group Type.</td>
</tr>
<tr>
<td>short_description</td>
<td>A small note about a Location Group to use at your discretion.</td>
</tr>
<tr>
<td>locationGroup Type ID</td>
<td>The ID of the Location Group Type.</td>
</tr>
</tbody>
</table>

**IBM Cloud Network discovered resources**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_network</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>VLAN number.</td>
</tr>
<tr>
<td>object_id</td>
<td>Network unique identifier.</td>
</tr>
<tr>
<td>access_port_vlan</td>
<td>VLAN number.</td>
</tr>
<tr>
<td>is_external</td>
<td>Network space value. The value is either:</td>
</tr>
<tr>
<td></td>
<td>· true if the network_space is “PUBLIC”</td>
</tr>
<tr>
<td></td>
<td>· false if the network_space is “PRIVATE”.</td>
</tr>
<tr>
<td>state</td>
<td>State has the value ‘available’.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_subnet</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>name</td>
<td>Name, composed of: &quot;network_identifier/cidr&quot;</td>
</tr>
<tr>
<td>object_id</td>
<td>Subnet unique identifier.</td>
</tr>
<tr>
<td>subnet_mask</td>
<td>Netmask of the subnet.</td>
</tr>
<tr>
<td>gateway</td>
<td>Gateway IP address.</td>
</tr>
<tr>
<td>cidr</td>
<td>Method for allocating IP addresses and IP routing: &quot;network_identifier/cidr&quot;.</td>
</tr>
<tr>
<td>category</td>
<td>Subnet type.</td>
</tr>
<tr>
<td>broadcast_address</td>
<td>Subnet broadcast address.</td>
</tr>
<tr>
<td>available_ip_count</td>
<td>Number of available IP addresses in the subnet.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_compute_security_group</td>
<td>Security Group name.</td>
</tr>
<tr>
<td>name</td>
<td>Security Group name.</td>
</tr>
<tr>
<td>object_id</td>
<td>Security Group unique identifier.</td>
</tr>
<tr>
<td>short_description</td>
<td>Security Group description that you provide for easy identification.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_compute_security_group_rule</td>
<td>Field with the same value as the Security Group Rule object_id field.</td>
</tr>
<tr>
<td>name</td>
<td>Field with the same value as the Security Group Rule object_id field.</td>
</tr>
<tr>
<td>object_id</td>
<td>Security Group Rule unique identifier.</td>
</tr>
<tr>
<td>created</td>
<td>Date the CI was created.</td>
</tr>
<tr>
<td>direction</td>
<td>Direction of the information. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· inbound</td>
</tr>
<tr>
<td></td>
<td>· outbound</td>
</tr>
<tr>
<td>Ethertype</td>
<td>Ethertype. One of the following:</td>
</tr>
<tr>
<td></td>
<td>· IPv4</td>
</tr>
<tr>
<td></td>
<td>· IPv6</td>
</tr>
<tr>
<td>port_range_min</td>
<td>The first port of the range that the rule applies to.</td>
</tr>
<tr>
<td>port_range_max</td>
<td>The last port of the range that the rule applies to.</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol that the rule is applied to.</td>
</tr>
<tr>
<td>remote_group_id</td>
<td>Security Group ID for the Source/Destination security group. It means that this rule applies on all IP addresses that were configured in this security group.</td>
</tr>
</tbody>
</table>

**IBM Cloud Organizations and Space discovered resources**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_cloud_org</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the cloud organization.</td>
</tr>
<tr>
<td>guid</td>
<td>Unique identifier of the organization.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>url</td>
<td>Unique URL of the organization.</td>
</tr>
</tbody>
</table>
| billing_enabled    | Billing method in use by the organization. One of either:  
  - Enabled / (TRUE)  
  - Disabled / (FALSE) |
| quota_definition_url | Resources that are available to an organization. |
| spaces_url         | Unique URL of the space assigned to the organization. |
| operational_status | Operational status of the organization. |
| name               | Unique name of the space. |
| object_id          | Space unique identifier (guid). |
| allow_ssh          | Access to the application over SSH is either allowed or not allowed. Boolean, one of either:  
  - TRUE  
  - FALSE |
| URL                | URL of the space from the metadata description. |

IBM Cloud Resource Groups discovered resources

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the resource group.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the resource group.</td>
</tr>
<tr>
<td>quota_id</td>
<td>Value identifying the quota ID associated with the resource group.</td>
</tr>
</tbody>
</table>

IBM Cloud SSH Key discovered resources

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name to identify an SSH key.</td>
</tr>
<tr>
<td>object_id</td>
<td>ID of the SSH key record.</td>
</tr>
<tr>
<td>key</td>
<td>SSH Key record.</td>
</tr>
<tr>
<td>finger_print</td>
<td>Short sequence of bytes used to authenticate or lookup a longer SSH Key. This value is automatically generated when adding or modifying the SSH Key. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>created_date</td>
<td>Date when the SSH Key was added. Read only. Changes made are ignored.</td>
</tr>
</tbody>
</table>
### ServiceNow

**New York**
**IT Operations Management**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>modify_date</td>
<td>Date when the SSH Key was modified. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>short_description</td>
<td>SSH description that you provide for easy identification.</td>
</tr>
<tr>
<td>name</td>
<td>Name of the datacenter.</td>
</tr>
<tr>
<td>region</td>
<td>Region that the datacenter is in.</td>
</tr>
<tr>
<td>discovery_status</td>
<td>Status information about the IBM datacenter.</td>
</tr>
</tbody>
</table>

**Related CI: cmdb_ci_ibm_datacenter**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the cloud service account.</td>
</tr>
<tr>
<td>account_id</td>
<td>Account ID for this cloud service account.</td>
</tr>
<tr>
<td>discovery_credentials</td>
<td>Credentials for this cloud service account.</td>
</tr>
<tr>
<td>datacenter_url</td>
<td>URL used to populate Logical Data Centers (LDCs).</td>
</tr>
</tbody>
</table>

**Related CI: cmdb_ci_cloud_service_account**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of a storage network resource.</td>
</tr>
<tr>
<td>object_id</td>
<td>Storage unique identifier.</td>
</tr>
<tr>
<td>size</td>
<td>Capacity of the storage account, measured in gigabytes.</td>
</tr>
<tr>
<td>size_bytes</td>
<td>Capacity of the storage account, measured in bytes.</td>
</tr>
<tr>
<td>label</td>
<td>Name of the storage network resource.</td>
</tr>
<tr>
<td>storage_type</td>
<td>Description of the storage type. As an example, NAS.</td>
</tr>
</tbody>
</table>

**IBM Cloud Storage discovered resources**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Username used to:</td>
</tr>
<tr>
<td></td>
<td>· access a non-EVault Storage volume</td>
</tr>
<tr>
<td></td>
<td>· register the EVault server agent with the vault backup system</td>
</tr>
<tr>
<td>object_id</td>
<td>Storage account unique identifier.</td>
</tr>
<tr>
<td>size</td>
<td>Capacity of the storage account, measured in gigabytes.</td>
</tr>
<tr>
<td>size_bytes</td>
<td>Capacity of the storage account, measured in bytes.</td>
</tr>
<tr>
<td>label</td>
<td>Name of the storage network resource.</td>
</tr>
<tr>
<td>storage_type</td>
<td>Description of the storage type. As an example, NAS.</td>
</tr>
<tr>
<td>fqdn</td>
<td>IP address of a storage resource or fully qualified domain name.</td>
</tr>
<tr>
<td>device</td>
<td>Unique identification number of the device associated with a Storage volume.</td>
</tr>
</tbody>
</table>

**Related CI: cmdb_ci_storage_vol_snapshot**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of a storage network resource.</td>
</tr>
<tr>
<td>object_id</td>
<td>Storage unique identifier.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### ServiceNow New York IT Operations Management

#### IBM Cloud Virtual Server discovered resources

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main CI: cmdb_ci_vm_instance</strong></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the VM.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the VM.</td>
</tr>
<tr>
<td>dns_domain</td>
<td>VM domain server name.</td>
</tr>
<tr>
<td>fqdn</td>
<td>VM IP address.</td>
</tr>
<tr>
<td>short_description</td>
<td>Description in a note that is attached to the VM on the IBM console.</td>
</tr>
<tr>
<td>cpus</td>
<td>Number of CPU cores assigned to this VM.</td>
</tr>
<tr>
<td>memory</td>
<td>Memory size assigned to this VM.</td>
</tr>
<tr>
<td>disks</td>
<td>Number of disks assigned to this VM.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>The running state of the VM.</td>
</tr>
<tr>
<td></td>
<td>• On – returned as ‘Running’ from the API.</td>
</tr>
<tr>
<td></td>
<td>• Off – returned as ‘Halted’ from the API.</td>
</tr>
<tr>
<td>placement_group_id</td>
<td>Placement group ID.</td>
</tr>
<tr>
<td>nics</td>
<td>Number of NICs.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_nic</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name, composed of:</td>
</tr>
<tr>
<td></td>
<td>• load balancer name</td>
</tr>
<tr>
<td></td>
<td>• Port</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the NIC.</td>
</tr>
<tr>
<td>ip_address</td>
<td>IP that is configured on the NIC.</td>
</tr>
<tr>
<td>mac_address</td>
<td>NIC MAC details.</td>
</tr>
<tr>
<td>netmask</td>
<td>Mask used to divide the IP address into subnets and to specify the available hosts in the network.</td>
</tr>
<tr>
<td>status</td>
<td>State of the NIC.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_endpoint_vnic</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name for the VNIC endpoint.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the VNIC endpoint.</td>
</tr>
<tr>
<td>host</td>
<td>Host identifier.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_storage_volume</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Username used to:</td>
</tr>
<tr>
<td></td>
<td>• access a non-EVault Storage volume</td>
</tr>
<tr>
<td></td>
<td>• register the EVault server agent with the vault backup system</td>
</tr>
<tr>
<td>object_id</td>
<td>Storage account unique identifier.</td>
</tr>
<tr>
<td>size</td>
<td>Capacity of the storage account, measured in gigabytes.</td>
</tr>
<tr>
<td>size_bytes</td>
<td>Capacity of the storage account, measured in bytes.</td>
</tr>
<tr>
<td>label</td>
<td>Name of the storage network resource.</td>
</tr>
<tr>
<td>storage_type</td>
<td>Description of the storage type. As an example, NAS.</td>
</tr>
<tr>
<td>fqdn</td>
<td>IP address of a storage resource or fully qualified domain name.</td>
</tr>
<tr>
<td>device</td>
<td>Unique identification number of the guest associated with a Storage volume.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_endpoint_nfs</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name for the NFS endpoint.</td>
</tr>
<tr>
<td>host</td>
<td>Host identifier.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol that was applied.</td>
</tr>
<tr>
<td>type</td>
<td>NFS endpoint type.</td>
</tr>
<tr>
<td>created</td>
<td>Date that the NFS endpoint was created.</td>
</tr>
<tr>
<td>updated</td>
<td>Date that the NFS endpoint was updated.</td>
</tr>
<tr>
<td>discovery_source</td>
<td>Source of the discovery. As an example, ServiceWatch</td>
</tr>
<tr>
<td>host_name</td>
<td>Host name of the NFS endpoint.</td>
</tr>
<tr>
<td>Related CI: cmdb_CI_endpoint_iscsi</td>
<td></td>
</tr>
<tr>
<td>host</td>
<td>Host identifier for the ISCSI endpoint.</td>
</tr>
<tr>
<td>port</td>
<td>Port to determine how GCP iscsi contact to the VMs.</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol that was applied.</td>
</tr>
<tr>
<td>type</td>
<td>ISCSI endpoint type</td>
</tr>
<tr>
<td>created</td>
<td>Date that the ISCSI endpoint was created.</td>
</tr>
<tr>
<td>updated</td>
<td>Date that the ISCSI endpoint was updated.</td>
</tr>
<tr>
<td>discovery_source</td>
<td>Source of the discovery. As an example, ImportSet</td>
</tr>
<tr>
<td>host_name</td>
<td>Host name of the ISCSI endpoint.</td>
</tr>
<tr>
<td>Related CI: cmdb_CI_endpoint_block</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name for the block endpoint.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the block endpoint.</td>
</tr>
<tr>
<td>host</td>
<td>Host identifier.</td>
</tr>
<tr>
<td>Related CI: cmdb_CI_os_template</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the image. For example, ‘CentOS 7.0-64 Minimal for VSI’.</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier of the image.</td>
</tr>
<tr>
<td>guest_os</td>
<td>OS used by the image. As an example, Linux.</td>
</tr>
<tr>
<td>root_device_type</td>
<td>Type of device used by root.</td>
</tr>
<tr>
<td>image_source</td>
<td>Source of the image.</td>
</tr>
<tr>
<td>image_type</td>
<td>Image type.</td>
</tr>
<tr>
<td>infuse_key</td>
<td>Key used to infuse the image.</td>
</tr>
<tr>
<td>update_host_name</td>
<td>Update host name for the image.</td>
</tr>
<tr>
<td>credentials</td>
<td>Credentials for the image.</td>
</tr>
<tr>
<td>version</td>
<td>OS version. For example, ‘7.0-64 Minimal for VSI’.</td>
</tr>
<tr>
<td>Related CI: cmdb_CI_cloud_key_pair</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name to identify a key (key label).</td>
</tr>
<tr>
<td>object_id</td>
<td>ID of the key record.</td>
</tr>
<tr>
<td>key</td>
<td>Key record.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>finger_print</td>
<td>Short sequence of bytes used to authenticate or look up a Key. This value is automatically generated when adding or modifying the Key. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>created_date</td>
<td>Date when the Key was added. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>modify_date</td>
<td>Date when the Key was modified. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>short_description</td>
<td>Description that you provide for easy identification.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_compute_security_group</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Security Group name.</td>
</tr>
<tr>
<td>object_id</td>
<td>Security Group unique identifier.</td>
</tr>
<tr>
<td>short_description</td>
<td>Security Group description that you provide for easy identification.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_endpoint_comp_security</td>
<td></td>
</tr>
<tr>
<td>host</td>
<td>Host identifier of the compute security endpoint.</td>
</tr>
<tr>
<td>port</td>
<td>Unique identifier of the compute security endpoint.</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol that was applied.</td>
</tr>
<tr>
<td>type</td>
<td>Type of compute security endpoint.</td>
</tr>
<tr>
<td>created</td>
<td>Date when the compute security endpoint was added. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>updated</td>
<td>Date when the compute security endpoint was modified. Read only. Changes made are ignored.</td>
</tr>
<tr>
<td>discovery_source</td>
<td>Source of the discovery. As an example, Manual Entry.</td>
</tr>
<tr>
<td>host_name</td>
<td>Host name of the compute security endpoint.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the datacenter.</td>
</tr>
<tr>
<td>region</td>
<td>Region that the datacenter is in.</td>
</tr>
<tr>
<td>discovery_status</td>
<td>Status information about the IBM datacenter.</td>
</tr>
<tr>
<td>sys_class_name</td>
<td>Class of the datacenter</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the availability zone (AZ).</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique identifier for this resource.</td>
</tr>
<tr>
<td>short_description</td>
<td>Availability zone description that you provide.</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_service_account</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the cloud service account.</td>
</tr>
<tr>
<td>account_id</td>
<td>Account ID for this cloud service account.</td>
</tr>
<tr>
<td>discovery_credentials</td>
<td>Credentials for this cloud service account.</td>
</tr>
<tr>
<td>datacenter_url</td>
<td>URL used to populate LDCs.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>datacenter_type</td>
<td>Type of datacenter. As an example, IBM Datacenter (cmdb_ci_ibm_datacenter)</td>
</tr>
<tr>
<td>datacenter_discovery_status</td>
<td>Status of last execution of Discover Datacenters.</td>
</tr>
</tbody>
</table>

### IBM Cloud Virtual Server – Related Items view

#### CI Relationships

These relationships are created to support IBM Cloud Platform discovery.

#### Relationships for IBM Cloud Load Balancer discovered resources

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
</tbody>
</table>
### Relationships for IBM Cloud Load Balancer discovered resources

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_cloud_load_balancer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_cloud_load_balancer</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_cloud_load_balancer</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_lb_health_service</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_lb_service</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_service</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_service</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_lb_pool</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_pool</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
<tr>
<td>cmdb_ci_lb_pool</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_lb_pool_member</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_lb_health_service</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_lb_health_service</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_cloud_load_balancer</td>
</tr>
</tbody>
</table>

### Relationships for IBM Cloud Location Groups discovered resources

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_location_group</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_cloud_location_group</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_cloud_location_group</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
</tbody>
</table>

### Relationships for IBM Cloud Network discovered resources

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_network</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_network</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_network</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_cloud_subnet</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_compute_security_group</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_compute_security_group</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>CI</td>
<td>Relationship</td>
<td>CI</td>
</tr>
<tr>
<td>----</td>
<td>--------------</td>
<td>----</td>
</tr>
<tr>
<td>cmdb_ci_compute_security_group</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_compute_security_group_rule</td>
</tr>
</tbody>
</table>

**Relationships for IBM Cloud Resource Groups discovered resources**

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_resource_group</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_resource_group</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
</tbody>
</table>

**Relationships for IBM Cloud SSH Key discovered resources**

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_cloud_key_pair</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_cloud_key_pair</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
</tbody>
</table>

**Relationships for IBM Cloud Storage discovered resources**

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_storage_volume</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_storage_volume</td>
<td>Provisioned::Provisioned From</td>
<td>cmdb_ci_storage_vol_snapshot</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_storage_vol_snapshot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_storage_vol_snapshot</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>cmdb_ci_storage_vol_snapshot</td>
<td>Located in Zone::Zone contains</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_availability_zone</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_availability_zone</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_storage_volume</td>
</tr>
</tbody>
</table>

**Relationships for IBM Cloud Virtual Server discovered resources**

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_ibm_datacenter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>cmdb_ci_ibm_datacenter</td>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_instance</td>
</tr>
</tbody>
</table>
### Related CI: cmdb_ci_vm_instance

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>Located in::Houses</td>
<td>cmdb_ci_availability_zone</td>
</tr>
<tr>
<td>Uses::Used By</td>
<td>cmdb_ci_cloud_key_pair</td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>cmdb_ci_nic</td>
</tr>
<tr>
<td>endpoint</td>
<td>cmdb_ci_nic</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_availability_zone

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_nic

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_os_template

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>Provisioned::Provisioned by</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_compute_security_group

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
<tr>
<td>Implement::Implemented from</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_storage_volume

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted on::Hosts</td>
<td>cmdb_ci_ibm_datacenter</td>
</tr>
<tr>
<td>Implement::Implemented from</td>
<td>cmdb_ci_vm_instance</td>
</tr>
</tbody>
</table>

### Related CI: cmdb_ci_cloud_key_pair

<table>
<thead>
<tr>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosted On::Hosts</td>
<td>cmdb_ci_cloud_service_account</td>
</tr>
</tbody>
</table>

---

**Troubleshooting**

If the mapping result is not as you expected, perform basic troubleshooting.
<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
</tbody>
</table>

**Note:** If the Configuration Parameters related list for the relevant MID Server does not show this parameter, you may need to add it.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout.</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the sa.debugger.max_timeout parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

**IBM Informix Dynamic Server discovery**

Discovery and Service Mapping can find and map the Informix Dynamic Server.

**Prerequisites**

**User**

Provide a user with permissions to run the following commands without elevated rights:

- `install_directory/bin/onstat -g dis`
  This command gets the status and configuration of Informix Dynamic server.
- `install_directory/bin/dbaccess sysmaster`
  This command runs an sql query against the master schema.

The same user must have read permission for the following directories:

- `install_directory + */etc/sqlhosts`
  Access to this directory is necessary to get information about the port, host, and service definition of the Informix Dynamic server.
- `/etc/services`
  Access to this directory is required to get information about the host service definition.

**Credentials**

Configure SSH credentials in the Credentials module of the ServiceNow platform for accessing the server hosting the Informix Dynamic server.
Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informix Instance (cmdb_ci_db_informix_instance)</td>
<td>The Informix instance attributes.</td>
</tr>
<tr>
<td>tcp_port</td>
<td></td>
</tr>
<tr>
<td>running_process</td>
<td></td>
</tr>
<tr>
<td>version</td>
<td></td>
</tr>
<tr>
<td>install_directory</td>
<td></td>
</tr>
<tr>
<td>config_file</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>Informix Catalog (cmdb_ci_db_informix_catalog)</td>
<td>The database included in the Informix instance.</td>
</tr>
<tr>
<td>database</td>
<td></td>
</tr>
</tbody>
</table>

CI Relationships

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship/Reference</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informix Instance (cmdb_ci_db_informix_instance)</td>
<td>Contains::Contained by</td>
<td>Informix Catalog (cmdb_ci_db_informix_catalog)</td>
</tr>
</tbody>
</table>

Top-down

During top-down discovery, Service Mapping discovers connections between the Informix Instance and the Informix Catalogs that the instance contains and displays them as an inclusion on the application service map.

IBM PowerHA Cluster (HACMP) discovery

Discovery uses the IBM PowerHA Cluster (HACMP) pattern to find IBM PowerHA SystemMirror for AIX (formerly HACMP) high-availability clusters on the AIX UNIX and Linux for IBM System p platforms.

Discovery uses this pattern to run horizontal discovery.

You can use this pattern on the ServiceNow platform using the London release or later.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

- In UNIX Classification (AIX), the Triggers probes related list, configure this pattern as an exploration that runs after the OS pattern. It can also run after the probes.
Pattern configuration in Triggers probes

- Make sure that the MID Server has access to the target host and can execute all the commands that are implemented in the pattern.
- Verify that the pattern is set to run against the server IP address or hostname.
- Configure a UNIX user with permissions to run all commands defined in the official IBM documentation, and the following commands on AIX:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lssrc -s clstrmgrES</td>
<td>Verify that the clstrmgrES process is running.</td>
</tr>
<tr>
<td>lslpp -l cluster.license</td>
<td>Get the license and version data of the cluster.</td>
</tr>
<tr>
<td>lslpp -l cluster.license</td>
<td>Get the software label version data of the cluster.</td>
</tr>
<tr>
<td>awk '{$1=$2=$3=&quot;&quot;; print $0)</td>
<td>Get the name and state of the resource group.</td>
</tr>
<tr>
<td>/usr/es/sbin/cluster/utilities/</td>
<td>Set the path to variable - '$util'</td>
</tr>
<tr>
<td>cltopinfo -c</td>
<td>Get the current state of the cluster.</td>
</tr>
<tr>
<td>/usr/es/sbin/cluster/utilities/</td>
<td>Get the ID of the cluster.</td>
</tr>
<tr>
<td>cltopinfo -n</td>
<td>Get the names of the nodes participating in the cluster.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/usr/es/sbin/cluster/clstat -o</td>
<td>Get the state of the nodes participating in the cluster.</td>
</tr>
<tr>
<td>/usr/es/sbin/cluster/utilities/cllsif</td>
<td>Get network IP/Non-IP interface data.</td>
</tr>
<tr>
<td>lsvg</td>
<td>Get information about the physical volumes of the cluster.</td>
</tr>
<tr>
<td>lsvg -o</td>
<td>Get information about the volume groups of the cluster.</td>
</tr>
<tr>
<td>lsattr -E -l inet0 -a hostname</td>
<td>awk '{print $2}'</td>
</tr>
<tr>
<td>/usr/es/sbin/cluster/utilities/clshowres</td>
<td>grep 'Concurrent Volume Group'</td>
</tr>
</tbody>
</table>

**Data collected by Discovery during horizontal discovery**

The pattern discovers data that helps you understand the cluster topology of your organization and the relationships between its components. The discovered data includes the following tables and fields.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cluster Resource Group</strong> (cmdb_ci_cluster_resource_group)**</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the resource group of the cluster.</td>
</tr>
<tr>
<td>server</td>
<td>The server on which the resource group resides. Example: HOST_NAME of the server.</td>
</tr>
<tr>
<td>resource_group_status</td>
<td>The status of the resource group of the cluster. Example: Online.</td>
</tr>
<tr>
<td>resource_group_type</td>
<td>The type of the resource group. Examples: Concurrent, non-concurrent.</td>
</tr>
<tr>
<td>cluster</td>
<td>The name of the cluster to which the resource group belongs.</td>
</tr>
<tr>
<td><strong>UNIX Cluster Resource</strong> (cmdb_ci_unix_cluster_resource)**</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the volume group of the cluster.</td>
</tr>
<tr>
<td>resource_type</td>
<td>The resource type (volume group).</td>
</tr>
<tr>
<td>resource_status</td>
<td>The status of the volume group. Examples: Online, Offline.</td>
</tr>
<tr>
<td>cluster</td>
<td>The name of the cluster to which the resource belongs.</td>
</tr>
<tr>
<td><strong>UNIX Cluster Node</strong> (cmdb_ci_unix_cluster_node)**</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the node that participates in the cluster.</td>
</tr>
<tr>
<td>node_state</td>
<td>The state of the node in the context of the CI state.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The IP addresses assigned to the node.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>node_status</td>
<td>The status of the node in the context of the resource group to which it belongs.</td>
</tr>
<tr>
<td>cluster</td>
<td>The name of the cluster to which the node belongs.</td>
</tr>
</tbody>
</table>

AIX Server (cmdb_ci_aix_server)

| serial_number  | The serial number of the AIX server host. |
| model_number   | The model number of the AIX host. |
| memory         | The memory size installed on the AIX host. |
| fqdn           | The fully qualified domain name of the AIX host. |
| os             | The type of the operating system of the AIX host. |
| os_version     | The version of the operating system. |
| start_date     | The start date of the machine. |
| cpu_vendor     | The manufacturer of the CPU. |

UNIX Cluster (cmdb_ci_unix_cluster)

| name           | The name of the cluster. |
| cluster_id     | The ID of the cluster. Example: 0123456789. |
| cluster_status | The current state of the cluster. Examples: ST_STABLE, NOT_CONFIGURED. |
| cluster_version| The version of the cluster. Examples: 5.5; 6.1; 7.2. |
| cluster_type   | The type of the cluster. Relevant only for versions 7.2 and above. Example: Stretched. |
| short_description | Description of the cluster (version, ID). |

Cluster Virtual IP (cmdb_ci_cluster_vip)

| ip_address     | Hostname/IP address of the server on which the cluster resides. (The IBM PowerHA SystemMirror/HACMP cluster almost never has a dedicated virtual IP address configured.) |
| cluster        | The name of the cluster. |

On the Dependency Views map you can see all discovered cluster CIs in your organization and the relationships between them.

In this example, UNIX Cluster Node represents a cluster of UNIX Clusters. UNIX Cluster Node gets resources from UNIX Cluster Resource and uses Cluster Virtual IP. UNIX Cluster Node contains three Cluster Resource Groups.
Dependency Views sample map

CI relationships

These relationships are created to support IBM HACMP/PowerHA cluster discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIX Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>Hosted on::Hosts</td>
<td>AIX Server (cmdb_ci_aix_server)</td>
</tr>
<tr>
<td>UNIX Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>Cluster of::Cluster</td>
<td>UNIX Cluster (cmdb_ci_unix_cluster)</td>
</tr>
<tr>
<td>UNIX Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>Contains::Contained by</td>
<td>Cluster Resource Group (cmdb_ci_cluster_resource_group)</td>
</tr>
<tr>
<td>Cluster Resource Group (cmdb_ci_cluster_resource_group)</td>
<td>Contains::Contained by</td>
<td>UNIX Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
</tr>
<tr>
<td>UNIX Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
<td>Defines resources for::Gets</td>
<td>UNIX Cluster Node (cmdb_ci_unix_cluster_node)</td>
</tr>
<tr>
<td>UNIX Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
<td>resources from</td>
<td>UNIX Cluster (cmdb_ci_unix_cluster)</td>
</tr>
<tr>
<td>Cluster Virtual IP (cmdb_ci_cluster_vip)</td>
<td>Uses::Used by</td>
<td>UNIX Cluster Node (cmdb_ci_unix_cluster_node)</td>
</tr>
</tbody>
</table>
IBM Virtualization and Hardware Management Console discovery

ServiceNow® Discovery can use the Hardware Management Console (HMC) to find and manage virtual IBM systems, such as IBM System i, IBM System p, IBM System z, and IBM Power Systems.

Discovery supports the following CIs:

- HMC server
- IBM frame
- Logical partition (LPAR) instance
  
  An LPAR is virtual machine equivalent of IBM.
- LPAR resource

Discovery is used on the HMC server and the individual AIX machines (LPARs).

Prerequisites

- Configure SSH credentials on a ServiceNow instance to allow using the following HMC commands:
  
  - Issyscfg
  - Lshmc
  - Lshwres

- Configure the SSH credentials on a ServiceNow instance to allow using the lparstat -i command on the AIX logical partition (LPAR).

  **Note:** The lparstat -i command is in addition to the commands required for AIX host discovery.

- Install the CMDB application.

Configure a discovery

To configure a discovery, do the following tasks:

1. Complete a discovery on the HMC IP address. Make sure this discovery completes successfully before you continue.
2. Complete a discovery on the LPARs (the AIX machines).

Data collected during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_hmc_server</td>
<td>Name of HMC</td>
</tr>
<tr>
<td>name</td>
<td>Name of HMC</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>frame_count</td>
<td>Number of frames managed for the HMC. A frame is the physical machine on which the LPARs reside. A frame is the equivalent of an ESX machine.</td>
</tr>
<tr>
<td>serial number</td>
<td>Unique identifier for the HMC</td>
</tr>
<tr>
<td>ip_address</td>
<td>IP address of the HMC</td>
</tr>
<tr>
<td>os_version</td>
<td>Operating system version</td>
</tr>
<tr>
<td>cpu_type</td>
<td>CPU type</td>
</tr>
<tr>
<td>cpu_speed</td>
<td>CPU speed</td>
</tr>
<tr>
<td>cpu_count</td>
<td>CPU count</td>
</tr>
<tr>
<td>cpu_core_count</td>
<td>CPU core count</td>
</tr>
<tr>
<td>ram</td>
<td>Memory RAM</td>
</tr>
<tr>
<td>cmdb_ci_ibm_frame</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of frame</td>
</tr>
<tr>
<td>serial number</td>
<td>Unique identifier for the frame</td>
</tr>
<tr>
<td>curr_avail_sys_proc_units</td>
<td>Current available processor units</td>
</tr>
<tr>
<td>configurable_sys_proc_units</td>
<td>Configurable processor units</td>
</tr>
<tr>
<td>installed_sys_proc_units</td>
<td>Installed processor units</td>
</tr>
<tr>
<td>curr_avail_sys_mem</td>
<td>Current available memory units</td>
</tr>
<tr>
<td>configurable_sys_mem</td>
<td>Configurable memory units</td>
</tr>
<tr>
<td>installed_sys_mem</td>
<td>Installed memory units</td>
</tr>
<tr>
<td>cmdb_ci_lpar_instance</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of LPAR instance</td>
</tr>
<tr>
<td>state</td>
<td>Operational state, translated as follows:</td>
</tr>
<tr>
<td></td>
<td>- Running translates to On.</td>
</tr>
<tr>
<td></td>
<td>- Not Activated and Not Available translate to Off.</td>
</tr>
<tr>
<td>serial_number</td>
<td>Serial number of the LPAR</td>
</tr>
<tr>
<td>object_id</td>
<td>Unique object identifier. This identifier is concatenated from the frame serial number and the LPAR identifier.</td>
</tr>
<tr>
<td>is_vio</td>
<td>Identifies the LPAR as a VIO server. A VIO server is the LPAR that manages the physical components for other LPARs, such as network interface cards (NICs) and optical drives.</td>
</tr>
<tr>
<td>vio_servers</td>
<td>Two scenarios:</td>
</tr>
<tr>
<td></td>
<td>- For client LPARs, identifies the VIO servers that manage the client LPARs.</td>
</tr>
<tr>
<td></td>
<td>- For VIO servers, identifies the Client LPARs the VIO servers manage.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>cmdb_ci_lpar_resource</td>
<td>All the attributes that the lparstat-i command retrieves are listed individually for the configuration item (CI).</td>
</tr>
<tr>
<td>high watermark</td>
<td>This dynamic value is the highest recorded value of the online virtual CPU attribute, as collected by the lparstat-i command.</td>
</tr>
<tr>
<td></td>
<td>However, because the value is dynamic, the results of a discovery could affect the value. During discovery one of the following may happen:</td>
</tr>
<tr>
<td></td>
<td>• If the value of the online virtual CPU is lower than the value collected by the lparstat-i command, the value does not change.</td>
</tr>
<tr>
<td></td>
<td>• If the value of the online virtual CPU is higher than the value collected by the lparstat-i command, the higher value is used.</td>
</tr>
<tr>
<td>node_name</td>
<td>Name of the instance, along with the attribute that refers to the LPAR instance CI.</td>
</tr>
<tr>
<td>cmdb_ci_processor_pool</td>
<td></td>
</tr>
<tr>
<td>pool_id</td>
<td>The processor pool ID.</td>
</tr>
<tr>
<td>lpar_ids</td>
<td>The LPAR ID in the discovered pool.</td>
</tr>
<tr>
<td>lpar_names</td>
<td>LPAR name in the discovered pool.</td>
</tr>
<tr>
<td>frame_name</td>
<td>The name of the frame to which the pools belongs.</td>
</tr>
</tbody>
</table>

The following is a dependency view of the relationships between the IBM frames, the LPAR instances, and the AIX servers.
Kubernetes discovery

Discovery Kubernetes and OpenShift components of the Kubernetes deployment using the Kubernetes pattern.

Discovery finds Kubernetes versions 1.5, 1.6, 1.7, 1.8, 1.9, and 1.10 using the Kubernetes pattern.

You can use this pattern on the Now Platform using Jakarta P9, Kingston P5, London P8, Madrid P2 or New York.
The Collect OpenShift info extension section of the Kubernetes pattern discovers the OpenShift components of the Kubernetes deployment. The OpenShift Build Config extension section is available from the October 2019 release, Store version 1.0.53. The OpenShift Build Config extension section is supported on the London patch 10, Madrid patch 5, and New York.

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store.

Prerequisites

Perform the following steps to ensure that Discovery can use the new pattern to successfully find Kubernetes.

1. On the Kubernetes platform, find the parameters to set up Kubernetes discovery:
   - Find the url of the kubeapi server:
     1. On the Kubernetes platform, run the following command:
     
        ```bash
        kubectl cluster-info
        ```
     2. In the output, find the line that states the url of the kubeapi server. For example,

        ```text
        https://10.154.144.146:443
        ```
   - Find the namespaces of the kubeapi server:
     1. On the Kubernetes platform, run this command:

        ```bash
        kubectl get namespaces
        ```
     2. In the output, find the line that states the namespaces. For example, `kube-system`.
   - Find out the Kubernetes username and password:
     1. On the Kubernetes platform, run this command:

        ```bash
        kubectl config view
        ```
     2. In the output, find the username and password.

```text
contexts:
- context:
  cluster: juju-cluster
  user: admin
  name: juju-context
  current-context: juju-context
kind: Config
preferences: {}
users:
- name: admin
  user:
    password: X3w8ZS1kSeFnbQI3i90Z67G
  username: admin
ruty@kube1:~$ Kubernetes password and username
```

- If you are planning to use advance authentication using a bearer token, find the bearer token:
1. On the Kubernetes platform, use the following command: `kubectl describe secret [default-token-g6pwc parameter]`,

   Where `kubectl` is the generic Kubernetes client,
   `describe` is the generic `kubectl` method,
   `secret` parameter reflects that in your Kubernetes cluster the default token is protected by containing it inside the object of type `secret`,
   `default-token-g6pwc` is the parameter value of the default token for `g6pwc`.

   The output shows all existing tokens for this Kubernetes account.

2. Choose one token to use for configuring Kubernetes credentials on the ServiceNow platform.

   2. Create the Kubernetes credentials:
      a. On the ServiceNow platform, navigate to **Discovery > Credentials**.
      b. Click **New**.
      c. Click **Kubernetes credentials**.
      d. Configure the form fields (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this credential.</td>
</tr>
<tr>
<td>User name</td>
<td>Enter the user name to create in the Credentials table. Avoid leading or</td>
</tr>
<tr>
<td></td>
<td>trailing spaces in user names. A warning appears if the platform detects</td>
</tr>
<tr>
<td></td>
<td>leading or trailing spaces in the user name.</td>
</tr>
<tr>
<td>Password</td>
<td>Enter the password for this credential.</td>
</tr>
<tr>
<td>Bearer Token</td>
<td><em>(Optional)</em> For advanced authentication using a bearer token, enter the</td>
</tr>
<tr>
<td></td>
<td>bearer token that Discovery use for authentication when accessing</td>
</tr>
<tr>
<td></td>
<td>Kubernetes. Use the character sequence as the token, for example, <code>31ada4fd-</code></td>
</tr>
<tr>
<td></td>
<td><code>adec-460c-809a-9e56ceb75269</code>.</td>
</tr>
<tr>
<td>Bearer Token Authentication</td>
<td><em>(Optional)</em> For advanced authentication using a bearer token, select this check box.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Credential alias      | 1. Click the padlock icon, and then click the search icon.  
2. On the Connection & Credential Aliases form, click New.  
3. Define attributes for the alias. Set the Type to Credential.  
4. Right-click the form header and click Save, then click Update.  
5. On the Connection & Credential Aliases form, click the newly added alias. The alias appears in the Credential alias field. |

**Note:** The alias allows you to use the Kubernetes credential for devices and applications other than Kubernetes.

e. On the Kubernetes credentials form, click **Update**.

3. Provide a user with the permissions to the following /api/v1 elements:
   - https://<url>/api/v1/namespaces/
   - https://<url>/api/v1/namespaces/<namespace>
   - https://<url>/api/v1/namespaces/kube-system/endpoints/kube-controller-manager
   - https://<url>/api/v1/services
   - https://<url>/api/v1/pods
   - https://<url>/api/v1/nodes

4. To discover the OpenShift components of the Kubernetes deployment, provide a user with the permissions to run the following /api/v1 elements:
   - /apis/apps.openshift.io/v1/deploymentconfigs
   - /apis/build.openshift.io/v1/buildconfigs
   - /apis/route.openshift.io/v1/routes
   - /apis/user.openshift.io/v1/groups
   - /apis/user.openshift.io/v1/users
   - /apis/project.openshift.io/v1/projects
   - /apis/image.openshift.io/v1/images
   - /apis/image.openshift.io/v1/imagestream

5. Create a serverless discovery schedule using the IP address and the port of the kubeapi server. Set the **Pattern** parameter to Kubernetes.

6. Create and define the serverless execution pattern as described in the product documentation. Configure the parameters required by the Kubernetes pattern as follows:
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host_ip</td>
<td>The identifier for the host_ip of the kubernetes apiserver.</td>
</tr>
<tr>
<td>port</td>
<td>The identifier for the kubernetes apiserver port.</td>
</tr>
</tbody>
</table>
| namespace   | Namespaces the system passed in the Kubernetes Discovery Configuration. Enter one of the following values:  
  - The actual namespace you found on the kubernetes platform. If there are more than one namespace, use comma (,) to separate values.  
  - Asterisk (*) for all namespaces.  
  - default" as the default value for namespaces. |
| credentialAlias | The alias associated with the previously created kubernetes credentials you created earlier. |

Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kubernetes Cluster (cmdb_ci_kubernetes_cluster)</td>
<td>The configuration item that represents the kubernetes deployment with all its component.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the kube-controller-manager leader.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The identifier for the host_ip of the kubernetes apiserver.</td>
</tr>
<tr>
<td>port</td>
<td>The identifier for the kubernetes apiserver port.</td>
</tr>
<tr>
<td>namespace</td>
<td>This value shows the namespaces the system passed in the Kubernetes Discovery Configuration.</td>
</tr>
<tr>
<td>Kubernetes Node (cmdb_ci_kubernetes_node)</td>
<td>The virtual aspect of the kubernetes node. Data relating to the physical aspect of the kubernetes node is stored under Linux server.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the kubernetes node. The format can be only the name of the machine or the full name consisting of the name and the hostname: &lt;name&gt;.&lt;hostname&gt; .</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The identifier for the kubernetes node UUID.</td>
</tr>
<tr>
<td>operational_status</td>
<td>The operational status of the kubernetes node.</td>
</tr>
<tr>
<td>Kubernetes Service (cmdb_ci_kubernetes_service)</td>
<td>The name of the kubernetes service.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>selector</td>
<td>This value shows the selector. The system uses this parameter to create the relation between the kubernetes service and the kubernetes pod.</td>
</tr>
<tr>
<td>namespace</td>
<td>The kubernetes namespace to which this kubernetes service belongs.</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes service UUID.</td>
</tr>
<tr>
<td>Kubernetes Pod (cmdb_ci_kubernetes_pod)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the Kubernetes pod.</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes pod UUID.</td>
</tr>
<tr>
<td>resourceVersion</td>
<td>The resource version of the Kubernetes pod.</td>
</tr>
<tr>
<td>namespace</td>
<td>The Kubernetes namespace to which this Kubernetes pod belongs.</td>
</tr>
<tr>
<td>Docker Container (cmdb_ci_docker_container)</td>
<td>The component that runs the docker image.</td>
</tr>
<tr>
<td>state</td>
<td>The state of the Kubernetes docker container.</td>
</tr>
<tr>
<td>container_id</td>
<td>The unique identifier for the Kubernetes docker container.</td>
</tr>
<tr>
<td>namespace</td>
<td>The kubernetes namespace to which this Kubernetes docker container belongs.</td>
</tr>
<tr>
<td>Docker Image (cmdb_ci_docker_image)</td>
<td>An executable package of an application and its related software that can be instantiated by a docker container.</td>
</tr>
<tr>
<td>image_id</td>
<td>The identifier for the Kubernetes docker image.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Kubernetes docker image.</td>
</tr>
<tr>
<td>image_url</td>
<td>The url for downloading the docker image.</td>
</tr>
<tr>
<td>namespace</td>
<td>The kubernetes namespace to which this Kubernetes docker image belongs.</td>
</tr>
<tr>
<td>Linux Server (cmdb_ci_linux_server)</td>
<td>The server that hosts the Kubernetes node.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Linux server powering the Kubernetes node.</td>
</tr>
<tr>
<td>hostname</td>
<td>The hostname of the Linux server.</td>
</tr>
<tr>
<td>serial_number</td>
<td>The serial number of the Linux server that serves as a unique identifier for the server.</td>
</tr>
<tr>
<td>cpu_type</td>
<td>The CPU architecture, for example 38-bit or 64-bit CPU.</td>
</tr>
<tr>
<td>cpu_count</td>
<td>The number of CPUs installed on this Linux server.</td>
</tr>
<tr>
<td>os</td>
<td>The operating system deployed on this Linux server.</td>
</tr>
<tr>
<td>kernel_release</td>
<td>The version of the Linux kernel operating system deployed on this Linux server.</td>
</tr>
<tr>
<td>ram</td>
<td>The size of RAM installed on this Linux server.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The IP address of the Linux server.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Key Value (cmdb_key_value)</td>
<td>This configuration item contains Kubernetes labels. Labels are key/value pairs that are attached to objects, such as pods.</td>
</tr>
<tr>
<td>key</td>
<td>The key of the kubernetes pod or Kubernetes service Key Value parameter.</td>
</tr>
<tr>
<td>value</td>
<td>The value of the kubernetes pod or Kubernetes service Key Value parameter.</td>
</tr>
<tr>
<td>Kubernetes Volume (cmdb_ci_kubernetes_volume)</td>
<td></td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes volume UUID.</td>
</tr>
<tr>
<td>mount_path</td>
<td>The path for accessing this Kubernetes volume.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Kubernetes volume.</td>
</tr>
<tr>
<td>namespace</td>
<td>The Kubernetes namespace to which this Kubernetes volume belongs.</td>
</tr>
<tr>
<td>volume_id</td>
<td>The ID of the Kubernetes volume.</td>
</tr>
<tr>
<td>OpenShift Deployed Configuration (cmdb_ci_openshift_dep_conf)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the OpenShift Deployment configuration.</td>
</tr>
<tr>
<td>namespace</td>
<td>The name of the namespace containing the deployment configuration.</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes volume UUID.</td>
</tr>
<tr>
<td>url</td>
<td>The URL of the OpenShift deployed configuration.</td>
</tr>
<tr>
<td>OpenShift Build Config (cmdb_ci_openshift_build_conf)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the OpenShift build configuration.</td>
</tr>
<tr>
<td>namespace</td>
<td>The name of the OpenShift namespace containing the build configuration.</td>
</tr>
<tr>
<td>k8s</td>
<td>The Kubernetes volume UUID.</td>
</tr>
<tr>
<td>url</td>
<td>The URL of the OpenShift build configuration.</td>
</tr>
<tr>
<td>OpenShift Source2Image (cmdb_ci_openshift_source_2_image)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the OpenShift source image.</td>
</tr>
<tr>
<td>to</td>
<td>Related Image.</td>
</tr>
<tr>
<td>parent_id</td>
<td>The ID of the OpenShift source image.</td>
</tr>
<tr>
<td>OpenShift Route (cmdb_ci_openshift_route)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the OpenShift route.</td>
</tr>
<tr>
<td>namespace</td>
<td>The name of the namespace containing the OpenShift route.</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes volume UUID.</td>
</tr>
<tr>
<td>url</td>
<td>The URL of the OpenShift Route.</td>
</tr>
<tr>
<td>host</td>
<td>The target host of the OpenShift route.</td>
</tr>
<tr>
<td>port</td>
<td>The target port of the OpenShift route.</td>
</tr>
<tr>
<td>OpenShift Group (cmdb_ci_openshift_group)</td>
<td></td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>name</td>
<td>The name of the OpenShift Group.</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes volume UUID.</td>
</tr>
<tr>
<td>url</td>
<td>The URL of the OpenShift Group.</td>
</tr>
</tbody>
</table>

OpenShift User (cmdb_ci_openshift_user)

| name            | The name of the OpenShift user. |
| k8s_uid         | The Kubernetes volume UUID. |
| url             | The URL of the OpenShift user. |
| full_name       | The full name of the OpenShift user. |

OpenShift Project (cmdb_ci_openshift_project)

| name            | The name of the OpenShift project. |
| k8s_uid         | The Kubernetes volume UUID. |
| url             | The URL of the OpenShift Project. |

OpenShift Image (cmdb_ci_openshift_images)

| name            | The name of the OpenShift Image. |
| k8s_uid         | The Kubernetes volume UUID. |
| url             | The URL of the OpenShift Image. |
| docker_image_metadata_id | The ID of the docker image. |
| docker_image_metadata_parent_id | The ID of the image parent ID. |
| arch            | Architecture of the image. |
| size            | The image size. |
| hostname        | The hostname related to the image. |

OpenShift Image Stream (cmdb_ci_openshift_images_stream)

| name            | The name of the OpenShift Image Stream. |
| k8s_uid         | The Kubernetes volume UID. |
| url             | The URL of the OpenShift Image Stream |
| namespace       | The name of the namespace containing the OpenShift image stream. |

OpenShift Docker Image Repository (cmdb_ci_openshift_docker_images_repository)

| name            | The name of the OpenShift docker image repository. |
| parent_ID       | The ID of the parent system. |

The graphic illustrates CIs that are part of Kubernetes discovery.

**Note:** This Dependency Views map was simplified for clarity. Your Kubernetes deployments can contain many more CIs.
CI relationships

These relationships are created to support Kubernetes discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kubernetes Cluster (cmdb_ci_kubernetes_cluster)</td>
<td>Contains::Contained By Kubernetes Service (cmdb_ci_kubernetes_service)</td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By Kubernetes Pod (cmdb_ci_kubernetes_pod)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By (cmdb_ci_openshift_source_2_image)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By OpenShift Group (cmdb_ci_openshift_group)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By OpenShift User (cmdb_ci_openshift_user)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By OpenShift Project (cmdb_ci_openshift_project)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By OpenShift Image (cmdb_ci_openshift_images)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Kubernetes event discovery

Discovery finds Kubernetes events and frequently updates the CMDB to reflect the dynamic Kubernetes environment. Discovery finds Kubernetes events for versions 1.5, 1.6, 1.7, 1.8, 1.9, and 1.10 using the Kubernetes Event pattern.

You can use this pattern on the ServiceNow platform using Jakarta P9, Kingston P5, London P8, or Madrid P2.

### Prerequisites

Perform the following steps to ensure that Discovery can use the new pattern to successfully find Kubernetes.

---

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contains::Contained By</td>
<td>OpenShift Docker Image Repository (cmdb_ci_openshift_docker_images_repository)</td>
<td></td>
</tr>
<tr>
<td>Cluster of::Cluster</td>
<td>Kubernetes Node (cmdb_ci_kubernetes_node)</td>
<td></td>
</tr>
<tr>
<td>Manages::Managed by</td>
<td>Linux Server (cmdb_ci_linux_server)</td>
<td></td>
</tr>
<tr>
<td>Kubernetes Service (cmdb_ci_kubernetes_service)</td>
<td>Cluster of::Cluster</td>
<td>Kubernetes Pod (cmdb_ci_kubernetes_pod)</td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>Docker Container (cmdb_ci_docker_container)</td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>Docker Image (cmdb_ci_docker_image)</td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>Kubernetes Volume (cmdb_ci_kubernetes_volume)</td>
<td></td>
</tr>
<tr>
<td>Docker Image (cmdb_ci_docker_image)</td>
<td>Instantiates::Instantiated by</td>
<td>Docker Container (cmdb_ci_docker_container)</td>
</tr>
<tr>
<td>Linux Server (cmdb_ci_linux_server)</td>
<td>Contains::Contained By</td>
<td>Kubernetes Pod (cmdb_ci_kubernetes_pod)</td>
</tr>
<tr>
<td>Runs::Runs on</td>
<td>Docker Container (cmdb_ci_docker_container)</td>
<td></td>
</tr>
<tr>
<td>Hosts::Hosted on</td>
<td>Kubernetes Node (cmdb_ci_kubernetes_node)</td>
<td></td>
</tr>
<tr>
<td>(cmdb_ci_kubernetes_namespace)</td>
<td>Contains::Contained By</td>
<td>OpenShift Deployed Config (cmdb_ci_openshift_deploy_conf)</td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>OpenShift Build Config (cmdb_ci_openshift_build_conf)</td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>OpenShift Route (cmdb_ci_openshift_route)</td>
<td></td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td>OpenShift Image Stream (cmdb_ci_openshift_images_stream)</td>
<td></td>
</tr>
<tr>
<td>OpenShift Deploy Config (cmdb_ci_openshift_deploy_conf)</td>
<td>Contains::Contained By</td>
<td>(cmdb_ci_config_file_tracked)</td>
</tr>
</tbody>
</table>
1. Discover Kubernetes in your organization using the Kubernetes pattern.

2. On the Kubernetes platform, find the parameters to set up Kubernetes discovery:
   - Find the url of the kubeapi server:
     1. On the Kubernetes platform, run the following command:
        ```bash
        kubectl cluster-info
        ```
     2. In the output, find the line that states the url of the kubeapi server. For example,
        ```bash
        https://10.154.144.146:443
        ```
   - Find the namespaces of the kubeapi server:
     1. On the Kubernetes platform, run this command:
        ```bash
        kubectl get namespaces
        ```
     2. In the output, find the line that states the namespaces. For example, `kube-system`.
   - If you are planning to use advance authentication using a bearer token, find the bearer token:
     1. On the Kubernetes platform, use the following command:
        ```bash
        kubectl describe secret [default-token-g6pwc parameter],
        ```
        Where `kubectl` is the generic Kubernetes client,
        `describe` is the generic `kubectl` method,
        `secret` parameter reflects that in your Kubernetes cluster the default token is protected by containing it inside the object of type `secret`,
        `default-token-g6pwc` is the parameter value of the default token for g6pwc.
        The output shows all existing tokens for this Kubernetes account.
     2. Choose one token to use for configuring Kubernetes credentials on the ServiceNow platform.

3. Create a serverless discovery schedule to run every 5 or 10 minutes.

   **Note:** When the pattern is run for the first time, it stores an event_timestamp. Later on it collects only the delta events based on the timestamp. The more often the pattern is run, the fewer updates to the CMDB IRE are needed.

4. Create a serverless execution pattern for the discovery schedule and assign it to the Kubernetes Events pattern. Configure the discovery pattern launcher the parameters required by the Kubernetes pattern as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>url</strong></td>
<td>The identifier for the hostname, IP or FQDN and the port of the Kubernetes api server. Use the following format: <code>example_hostname:example_port</code> or <code>example_ip:example_port</code>.</td>
</tr>
</tbody>
</table>
## Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| namespace      | Namespaces the system passed in the Kubernetes Discovery Configuration. Enter one of the following values:  
- The actual namespace you found on the Kubernetes platform. If there is more than one namespace, use comma (,) to separate values.  
- Asterisk (*) for all namespaces.  
- default" as the default value for namespaces. |
| credentialsAlias | The alias associated with the previously created Kubernetes credentials. |

### Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kubernetes Cluster (cmdb_ci_kubernetes_cluster)</td>
<td>The configuration item that represents the kubernetes deployment with all its component.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the kube-controller-manager leader.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The identifier for the host_ip of the kubernetes apiserver.</td>
</tr>
<tr>
<td>port</td>
<td>The identifier for the kubernetes apiserver port.</td>
</tr>
<tr>
<td>namespace</td>
<td>This value shows the namespaces the system passed in the Kubernetes Discovery Configuration.</td>
</tr>
<tr>
<td>event_timestamp</td>
<td>The timestamp of the latest event created on this Kubernetes cluster at the time of the discovery.</td>
</tr>
<tr>
<td>Kubernetes Node (cmdb_ci_kubernetes_node)</td>
<td>The virtual aspect of the kubernetes node. Data relating to the physical aspect of the kubernetes node is stored under Linux server.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the kubernetes node. The format can be only the name of the machine or the full name consisting of the name and the hostname: <code>&lt;name&gt;</code>.<code>&lt;hostname&gt;</code> .</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The identifier for the kubernetes node UUID.</td>
</tr>
<tr>
<td>operational_status</td>
<td>The operational status of the kubernetes node.</td>
</tr>
<tr>
<td>Kubernetes Service (cmdb_ci_kubernetes_service)</td>
<td>The name of the kubernetes service.</td>
</tr>
<tr>
<td>name</td>
<td>This value shows the selector. The system uses this parameter to create the relation between the kubernetes service and the kubernetes pod.</td>
</tr>
<tr>
<td>namespace</td>
<td>The kubernetes namespace to which this kubernetes service belongs.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes service UUID.</td>
</tr>
<tr>
<td>Kubernetes Pod (cmdb_ci_kubernetes_pod)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of the Kubernetes pod.</td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes pod UUID.</td>
</tr>
<tr>
<td>resourceVersion</td>
<td>The resource version of the Kubernetes pod.</td>
</tr>
<tr>
<td>namespace</td>
<td>The Kubernetes namespace to which this Kubernetes pod belongs.</td>
</tr>
<tr>
<td>Docker Container (cmdb_ci_docker_container)</td>
<td>The component that runs the docker image.</td>
</tr>
<tr>
<td>state</td>
<td>The state of the Kubernetes docker container.</td>
</tr>
<tr>
<td>container_id</td>
<td>The unique identifier for the Kubernetes docker container.</td>
</tr>
<tr>
<td>namespace</td>
<td>The kubernetes namespace to which this Kubernetes docker container belongs.</td>
</tr>
<tr>
<td>Docker Image (cmdb_ci_docker_image)</td>
<td>An executable package of an application and its related software that can be instantiated by a docker container.</td>
</tr>
<tr>
<td>image_id</td>
<td>The identifier for the Kubernetes docker image.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Kubernetes docker image.</td>
</tr>
<tr>
<td>image_url</td>
<td>The url for downloading the docker image.</td>
</tr>
<tr>
<td>namespace</td>
<td>The kubernetes namespace to which this Kubernetes docker image belongs.</td>
</tr>
<tr>
<td>Linux Server (cmdb_ci_linux_server)</td>
<td>The server that hosts the Kubernetes node.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Linux server powering the Kubernetes node.</td>
</tr>
<tr>
<td>hostname</td>
<td>The hostname of the Linux server.</td>
</tr>
<tr>
<td>serial_number</td>
<td>Enter the serial number of the Linux server that serves as a unique identifier for the server.</td>
</tr>
<tr>
<td>cpu_type</td>
<td>The CPU architecture, for example 38-bit or 64-bit CPU.</td>
</tr>
<tr>
<td>cpu_count</td>
<td>The number of CPUs installed on this Linux server.</td>
</tr>
<tr>
<td>os</td>
<td>The operating system deployed on this Linux server.</td>
</tr>
<tr>
<td>kernel_release</td>
<td>The version of the Linux kernel operating system deployed on this Linux server.</td>
</tr>
<tr>
<td>ram</td>
<td>The size of RAM installed on this Linux server.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The IP address of the Linux server.</td>
</tr>
<tr>
<td>Key Value (cmdb_key_value)</td>
<td>This configuration item contains Kubernetes labels. Labels are key/value pairs that are attached to objects, such as pods.</td>
</tr>
<tr>
<td>key</td>
<td>The key of the kubernetes pod or Kubernetes service Key Value parameter.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>value</td>
<td>The value of the kubernetes pod or Kubernetes service Key Value parameter.</td>
</tr>
<tr>
<td>Kubernetes Volume (cmdb_ci_kubernetes_volume)</td>
<td></td>
</tr>
<tr>
<td>k8s_uid</td>
<td>The Kubernetes volume UUID.</td>
</tr>
<tr>
<td>mount_path</td>
<td>The path for accessing this Kubernetes volume.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the Kubernetes volume.</td>
</tr>
<tr>
<td>namespace</td>
<td>The Kubernetes namespace to which this Kubernetes volume belongs.</td>
</tr>
<tr>
<td>volume_id</td>
<td>The ID of the Kubernetes volume.</td>
</tr>
</tbody>
</table>

The graphic illustrates CIs that are part of Kubernetes discovery.

**Note:** This Dependency Views map was simplified for clarity. Your Kubernetes deployments can contain many more CIs.
CI relationships

These relationships are created to support Kubernetes event discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kubernetes Cluster (cmdb_ci_kubernetes_cluster)</td>
<td>Contains::Contained By</td>
<td>Kubernetes Service (cmdb_ci_kubernetes_service)</td>
</tr>
<tr>
<td>Contains::Contained By</td>
<td></td>
<td>Kubernetes Pod (cmdb_ci_kubernetes_pod)</td>
</tr>
</tbody>
</table>
Microsoft Azure Application Gateway discovery

Discovery uses the Azure Application Gateway HD pattern for discovering this product, while Service Mapping discovers Application Gateway using Azure Application Gateway TD (LBS) pattern.

Discovery and Service Mapping discover the 2017-03-01 version of Azure Application Gateway. You can use this pattern on the ServiceNow platform using Jakarta Patch 10, Kingston Patch 8, or London Patch 1.

Prerequisites

Azure Principal credentials

On the ServiceNow platform, create a service account for cloud discovery. Set the account Id to the Azure Service Principal as it appears at the Microsoft Azure Console.

Privileges

On the Microsoft Azure Console, provide a user a user with the reader role.

Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Load Balancer (cmdb_ci_cloud_load_balancer)</td>
<td>The name of the Azure Application Gateway.</td>
</tr>
<tr>
<td>name</td>
<td>The fully qualified domain name of the Azure Application Gateway.</td>
</tr>
</tbody>
</table>
### Table and field

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object_id</td>
<td>The ID of the gateway.</td>
</tr>
<tr>
<td>Install_status</td>
<td>The install status of the Azure Application Gateway.</td>
</tr>
<tr>
<td>state</td>
<td>The state of the Azure Application Gateway.</td>
</tr>
<tr>
<td>Cloud LB IPAddress (cmdb_ci_cloud_lb_ipaddress)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The IP address of the Azure Application Gateway.</td>
</tr>
<tr>
<td>Object_id</td>
<td>The ID of the Azure Application Gateway.</td>
</tr>
<tr>
<td>ipaddress _type</td>
<td>The type of the IP address, which can be private or public.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The IP address of the Azure Application Gateway.</td>
</tr>
<tr>
<td>state</td>
<td>The state of the IP address of the Azure Application Gateway.</td>
</tr>
<tr>
<td>operational_state</td>
<td>The operational state of the IP address of the Azure Application Gateway.</td>
</tr>
<tr>
<td>install_status</td>
<td>The install status of the IP address of the Azure Application Gateway.</td>
</tr>
<tr>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td></td>
</tr>
<tr>
<td>serial number</td>
<td>The ID of the Azure Application Gateway service.</td>
</tr>
<tr>
<td>port</td>
<td>The TCP port that the Azure Application Gateway service listens to.</td>
</tr>
<tr>
<td>name</td>
<td>The FQDN with the port number of the Azure Application Gateway service, following this syntax: <code>&lt;fqdn&gt;:&lt;port&gt;</code>.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The IP address of the Azure Application Gateway on which the service runs.</td>
</tr>
<tr>
<td>object_id</td>
<td>The ID of the Azure Application Gateway service.</td>
</tr>
<tr>
<td>operational_state</td>
<td>The operational state of the Azure Application Gateway service.</td>
</tr>
</tbody>
</table>

### CI relationships

These relationships are created to support Azure Application Gateway discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
<td>Hosted on::Hosts</td>
<td>Cloud Load Balancer (cmdb_ci_cloud_load_balancer)</td>
</tr>
<tr>
<td>Cloud Load Balancer (cmdb_ci_cloud_load_balancer)</td>
<td>Owns::Owned by</td>
<td>Cloud LB IPAddress (cmdb_ci_cloud_lb_ipaddress)</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Connections discovered by Service Mapping during the top-down discovery

Service Mapping performs the top-down discovery of the Azure Application Gateway in the context of application services. Service Mapping discovers the outgoing http connections from Azure Application Gateway service to pool members, such as virtual machines.

Microsoft Azure Inventory discovery

Discovery uses the Azure Inventory pattern for discovering resources available through Azure. Use this pattern for Azure resources, which do not have a dedicated pattern like Azure Functions or Azure Application Gateway.

You can discover up to 10 thousand resources using the Azure Inventory pattern.

You can use this pattern on the ServiceNow platform using London (Patch 8) or later releases.

Request apps on the Store

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store. For cumulative release notes information for all released apps, refer to the ServiceNow Store version history release notes.

Prerequisites

Azure service account
 Configure the Azure service account.

Azure Principal credentials
 On the Now Platform, create a service account for cloud discovery. Set the account Id to the Azure Service Principal as it appears at the Microsoft Azure Console.

Privileges
 On the Microsoft Azure Console, provide a user with the API Management Service Reader role with permissions to run the following REST request:

```
href="https://management.azure.com/subscriptions/<subscription id>/resources"
format="html" scope="external"/>
```

Upgrade the Now Platform
 Upgrade the Now Platform to the Kingston release or later.

Logical Datacenters
 Discover Logical Datacenters hosting Azure Functions as described in Run Discovery on an Azure cloud service account in the ServiceNow Cloud Management documentation.

Create discovery schedule
 Create a schedule for the relevant Azure service account as described in Schedule Discovery on a service account in the ServiceNow Cloud Management documentation. Alternatively, create a schedule for running the Azure Resource Inventory pattern as described in Run cloud application discovery in the ServiceNow Discovery documentation.

Fine-tune Azure inventory discovery using the Resource Whitelist
 If your deployment has custom patterns for Azure discovery, ensure that you do not discover Azure resources twice.
1. Ensure that the application scope is Discovery and Service Mapping Patterns:
   a. Navigate to Settings > Developer.
   b. Select Discovery and Service Mapping Patterns from the Application list.

2. Navigate to System Definitions > Tables.
3. Open the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table.
4. Under Related Links, click Show List.
5. Select resource types for which you have custom patterns, and select Delete from the Actions on selected rows list.

   The Cloud Inventory Resource Whitelist is predefined with common services. You can expand the whitelist with additional resource types that you want the pattern to discover. The names of these resource types must conform to the appropriate vendor naming conventions.

   **Note:** When you modify the out-of-the-box whitelist, it is no longer updated automatically in application updates. You need to maintain the customized list on your own.

1. Open the Cloud Inventory Resource Whitelist (sa_cloud_inventory_resource_whitelist) table.
2. Click New.
3. Fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Vendor</td>
<td>The vendor of the resource type: Azure.</td>
</tr>
<tr>
<td>Resource Type</td>
<td>The AWS resource type value.</td>
</tr>
<tr>
<td>Application</td>
<td>The application scope: Discovery and Service Mapping Patterns.</td>
</tr>
</tbody>
</table>

The changes are applied the next time you run the pattern.

**Modify the ServiceNow identification rules.**

Be sure to switch to the Cloud Management Platform application scope before you begin the following procedure.

1. Navigate to CI Class Manager > Hierarchy > Cloud Resource > Identification Rule.
2. Delete and add the appropriate identification rule.

   The new identification rule should include only the Object ID attribute.
Modify the ServiceNow related entries

1. Navigate to Related Entries.
2. Click New.
3. For the identifier, enter `cmdb_ci_cmp_resource`.
4. For the related table, select Key Value (cmdb_key_value).
5. For the referenced field, select Configuration Item.
6. For the criterion attributes, add Key and Value.

You may need to click the lock icon to view and change the criterion attributes.
Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Resource (cmdb_ci_cmp_resource)</td>
<td>A descriptive name used to identify the Azure resource.</td>
</tr>
<tr>
<td>name</td>
<td></td>
</tr>
<tr>
<td>resource_type</td>
<td>The type of the resource, for example network, load balancer.</td>
</tr>
<tr>
<td>Object_id</td>
<td>The ID of Azure resource.</td>
</tr>
<tr>
<td>Description</td>
<td>The text populated during discovery that specifies that this resource is available from Azure.</td>
</tr>
<tr>
<td>Key Value (cmdb_key_value)</td>
<td></td>
</tr>
<tr>
<td>tag</td>
<td>The tag value that Discovery collects from the Azure resource.</td>
</tr>
</tbody>
</table>

CI relationships

These relationships are created to support Azure Functions discovery:

| CI | Relationship | CI |
|----------------|--------------|-----------------|-----------------|-----------------|
| Azure Resource (cmdb_ci_cmp_resource) | Hosted on:Hosts | Logical Datacenter (cmdb_ci_logical_datacenter) |

Troubleshooting

If the mapping result is not as you expected, perform basic troubleshooting.
### Microsoft Azure Functions discovery

Discovery and Service Mapping uses the Microsoft Azure Functions pattern for discovering Azure Functions service and mapping this service in the context of application services.

You can use this pattern on the Now Platform® using Jakarta Patch 10, Kingston Patch 8, or London Patch 1.

### Prerequisites

**Azure service account**

Configure the Azure service account.

**Azure Principal credentials**

On the Now Platform, create a service account for cloud discovery. Set the account Id to the Azure Service Principal as it appears at the Microsoft Azure Console.

**Privileges**

On the Microsoft Azure Console, provide a user with the API Management Service Reader role.

**Upgrade the Now Platform**

Upgrade the Now Platform to the relevant patch: Kingston Patch 8 or later.

**Note:** Patterns in the earlier patches do not support top-down discovery performed by Service Mapping.

**Logical Datacenters**

Discover Logical Datacenters hosting Azure Functions as described in Run Discovery on an Azure cloud service account in the ServiceNow Cloud Management documentation.
Create discovery schedule

Create a schedule for the relevant Azure service account for running the as described in Schedule Discovery on a service account in the ServiceNow Cloud Management documentation.

Configure Azure Alert service

To enable alert-based discovery, configure Discovery and Service Mapping to trigger the pattern to discover Azure Functions based on Azure alerts:

1. In the Cloud Resource Types (sn_capi_resource_type) table, associate the Microsoft Azure Functions pattern with the Microsoft.Web/sites cloud resource type.
2. Configure the Azure Alert service as described in Set up Azure alert processing for Discovery and Service Mapping in the ServiceNow Cloud Management documentation.

Commands

The Azure Functions pattern contains the following commands that do not require elevated rights:

- (On Linux) egrep — This command searches text in the previous command output.
- (On Windows) findstr — Finds a string in the previous command output.

Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azure Function (cmdb_ci_cloud_function)</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>The name of Azure Functions.</td>
</tr>
<tr>
<td>short_description</td>
<td>The description that distinguishes the Azure Function from other cloud functions. The value is Azure Function.</td>
</tr>
<tr>
<td>Object_id</td>
<td>The ID of Azure Functions.</td>
</tr>
<tr>
<td>state</td>
<td>The state of Azure Functions.</td>
</tr>
<tr>
<td>version</td>
<td>The version of Azure Functions.</td>
</tr>
<tr>
<td>code location URL</td>
<td>The URL of the Azure Functions source code.</td>
</tr>
<tr>
<td>FQDN</td>
<td>The FQDN of the Azure Functions application.</td>
</tr>
</tbody>
</table>

CI relationships

These relationships are created to support Azure Functions discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azure Functions (cmdb_ci_cloud_function)</td>
<td>Hosted on:Hosts</td>
<td>Logical Datacenter (cmdb_ci_logical_datacenter)</td>
</tr>
<tr>
<td></td>
<td>Uses:Used by</td>
<td>Cloud Database (cmdb_ci_cloud_database)</td>
</tr>
</tbody>
</table>
Connections found by Service Mapping during top-down discovery

Service Mapping can discover the following connections:

- Connections to the following databases: MSSql, MySql, MongoDB, and Documentum.
- HTTP connections belonging to Azure Functions environment, for example, https://scenario4-hop1-azure-function.azurewebsites.net.

**Note:** Service Mapping does not scan java classes/jars and binary files for possible information about outgoing HTTP connections of Azure Functions. If java classes/jars and binary files are the only source of information about some outgoing HTTP connections, Service Mapping does not find such HTTP connections.

Troubleshooting

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout.</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the sa.debugger.max_timeout parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

Nutanix Acropolis discovery

Discovery uses the Nutanix Components pattern to find components of the Nutanix Acropolis solution containing Nutanix Prism Central version 5.10 or Nutanix Prism Element 5.9.
You can use this pattern on the Now Platform using Madrid Patch 5 and London Patch 9. The Nutanix Components pattern uses REST APIs to collect information about the Acropolis deployment components.

Prerequisites

MID Server requirements
Deploy and connect the dedicated MID Server for accessing the Nutanix deployment.

Credentials
Configure the following credentials:
1. Navigate to Discovery > Credentials.
2. Click New.
3. Click Basic Auth Credentials, fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Credential name. An example is Acropolis</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this credential for discovery. Select this check box to enable discovery.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Select All MID servers.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which the platform tries this credential as it attempts to log on to devices. A smaller number indicates that the credential appears higher in the list. Establish the credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number, or none, the instance tries the credentials in a random order. The default value is 100.</td>
</tr>
<tr>
<td>User name</td>
<td>Name of the user for performing Acropolis discovery.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for this user.</td>
</tr>
<tr>
<td>Credential alias</td>
<td>Credential alias for this credential to authorize actions.</td>
</tr>
</tbody>
</table>

Credentials for API elements
During the discovery, the pattern uses the following API elements. The user that was added to the credentials in the instance must have permissions to send these queries.

- /PrismGateway/services/rest/v2.0/tasks/list
- /api/nutanix/v3/clusters/list
- /PrismGateway/services/rest/v2.0/hosts
- /PrismGateway/services/rest/v2.0/clusters
- /PrismGateway/services/rest/v2.0/vms

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc. in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
(Optional) Create a discovery schedule

Create a discovery schedule to perform targeted discovery of Nutanix components. An alternative is to perform discovery using the HTTP classification, which queries all servers with open HTTP ports and may lead to overloading the network.

1. Navigate to Discovery > Discovery Schedules.
2. Click New and then fill in the form.

**Discovery Schedule form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the Discovery schedule. For example, Acropolis.</td>
</tr>
<tr>
<td>Discover</td>
<td>Discover type. Select <strong>Serverless</strong>.</td>
</tr>
<tr>
<td>MID Server</td>
<td>Name of the MID Server that can access Nutanix PRISM Central.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this schedule for discovery. Select this check box to enable discovery.</td>
</tr>
</tbody>
</table>

3. Right-click the header of the Discovery Schedule form and select **Save**.
4. Click the **Serverless Execution Patterns** tab, click **New**, and then fill in the form.

**Related Links**

Quick ranges
Discover now

**Serverless Execution Pattern form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for this Serverless Execution Pattern. For example, Acropolis.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern</td>
<td>Name of the pattern to run: Nutanix Components.</td>
</tr>
<tr>
<td>Proxy Host</td>
<td>Fully qualified domain name of the machine on which you are installing the proxy server. Specify Global.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this schedule for discovery. Select this check box to enable discovery.</td>
</tr>
</tbody>
</table>

5. Under **Discovery Pattern Launcher Parameters**, configure the IP address of the Prism Central.

**Optional** Enable the HTTP classification for using Quick Discovery

If you don’t want to create a serverless discovery schedule, you can trigger discovery of the Nutanix components with the HTTP classification. This classification enables you to run the discovery using Quick Discovery. In this case, the discovery runs on a single IP address without requiring a schedule.

To enable HTTP classification, ensure that the application scope is Discovery and Service Mapping Patterns:

1. Navigate to **Settings > Developer**.
2. Select **Discovery and Service Mapping Patterns** from the **Application** list.

Since the HTTP classification triggers queries of all servers with the open HTTP ports, it may lead to overloading the network and slowing down the discovery process.

1. Navigate to **Discovery > Discovery Definition > CI Classification > HTTP**.
2. Modify the filter to display all classifications including classifications that are not active.
3. Locate the **Nutanix HTTP** classification and set the **Active** attribute to **true**.

**CMDB CI Class Models store app**

Ensure that your instance supports the Nutanix CI class models by deploying the latest version of the CMDB CI Class Models store app from the ServiceNow Store.

**Optional** Discover VMware components

To enable viewing relations between Nutanix Acropolis and VMware, run VMware discovery before Nutanix Acropolis discovery. Run the horizontal discovery for the VMware Vcenter using the VMware - vCenter Datacenters probe.

**Optional** Disable CI tag discovery

The Nutanix Components pattern discovers tags for Nutanix hosts and virtual machines. If you have dozens of tags in your deployment, it may slow down the discovery causing performance issues. You can disable the CI tag discovery by performing the following steps:

1. Navigate to **Pattern Designer > Discovery Patterns**.
2. Select the **Nutanix Components** pattern.
3. Under **Extension Section**, select the check box for **Categories (Tags)** and click **Remove**.
4. Click **Save**.
## Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nutanix Controller VM (cmdb_ci_nutanix_controller_vm) - A Nutanix controller virtual machine that is present in each node and that provides the storage clustering and management capabilities.</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix Controller VM.</td>
</tr>
<tr>
<td>VM ID</td>
<td>The universally unique identifier (UUID) of the Nutanix Controller VM.</td>
</tr>
<tr>
<td>Object ID</td>
<td>The ID of the Nutanix Controller VM.</td>
</tr>
<tr>
<td>Memory</td>
<td>The memory size of the Nutanix Controller VM in Megabytes (MB).</td>
</tr>
<tr>
<td>State</td>
<td>The power state of the Nutanix Controller VM (on/off).</td>
</tr>
<tr>
<td>Hypervisor Type</td>
<td>The type of the hypervisor used for the Nutanix Controller VM.</td>
</tr>
<tr>
<td>Description</td>
<td>The description defined for the virtual machine in the Nutanix console: Prism Central or Prism Element.</td>
</tr>
<tr>
<td>Nutanix Cluster (cmdb_ci_nutanix_cluster) - The cluster made up of the physical nodes running Nutanix software.</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix Cluster.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address of the Nutanix Cluster.</td>
</tr>
<tr>
<td>Cluster ID</td>
<td>The ID of the Nutanix Cluster.</td>
</tr>
<tr>
<td>Object ID</td>
<td>The object ID of the Nutanix Cluster.</td>
</tr>
<tr>
<td>Version</td>
<td>The Nutanix Cluster version.</td>
</tr>
<tr>
<td>Full Version</td>
<td>The full version of the Nutanix Cluster.</td>
</tr>
<tr>
<td>Internet Subnet</td>
<td>The subnet of the internal IP address of the Nutanix Cluster.</td>
</tr>
<tr>
<td>External Subnet</td>
<td>The subnet of the external IP address of the Nutanix Cluster.</td>
</tr>
<tr>
<td>Timezone</td>
<td>The timezone of the Nutanix Cluster.</td>
</tr>
<tr>
<td>Hypervisor Types</td>
<td>The type of the hypervisor used for the Nutanix Cluster.</td>
</tr>
<tr>
<td>NTP Servers</td>
<td>The Network Time Protocol (NTP) servers that are related to the Nutanix Cluster.</td>
</tr>
<tr>
<td>Number of Nodes</td>
<td>The number of nodes connected to the Nutanix Cluster.</td>
</tr>
<tr>
<td>NCC Version</td>
<td>The Nutanix Cluster Check (NCC) version.</td>
</tr>
<tr>
<td>Block Serial Numbers</td>
<td>The serial numbers of blocks of data connected to the Nutanix Cluster.</td>
</tr>
<tr>
<td>Nutanix Host (cmdb_ci_nutanix_host) - The physical server hosting all the virtual machines.</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix host.</td>
</tr>
<tr>
<td>Object ID</td>
<td>The ID of the Nutanix host.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CPU Core Count</td>
<td>The number of Nutanix host cores.</td>
</tr>
<tr>
<td>CPU Core Thread</td>
<td>The number of threads of the Nutanix host.</td>
</tr>
<tr>
<td>CPU Speed</td>
<td>The CPU speed of the Nutanix host in MHz.</td>
</tr>
<tr>
<td>CPU Count</td>
<td>The number of CPU sockets of the Nutanix host.</td>
</tr>
<tr>
<td>RAM</td>
<td>The memory of the host in Megabites (MB).</td>
</tr>
<tr>
<td>Nutanix Storage Pool</td>
<td>Grouping of physical disks within a Nutanix cluster used to create a physical separation between virtual machines.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix storage pool.</td>
</tr>
<tr>
<td>Pool ID</td>
<td>The UUID of the Nutanix storage pool.</td>
</tr>
<tr>
<td>Object ID</td>
<td>The ID of the Nutanix storage pool.</td>
</tr>
<tr>
<td>Size</td>
<td>The size of the Nutanix storage pool.</td>
</tr>
<tr>
<td>Nutanix Storage Container</td>
<td>A subset of Nutanix storage pool used to apply policies such as reserved capacity, replication factor, and storage optimization options.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix container.</td>
</tr>
<tr>
<td>Replication Factor</td>
<td>The number of copies created for the data stored on this Nutanix storage container.</td>
</tr>
<tr>
<td>Deduplication</td>
<td>The deduplication is enabled or disabled for the Nutanix container.</td>
</tr>
<tr>
<td>Compression</td>
<td>The compression is enabled or disabled for the Nutanix container.</td>
</tr>
<tr>
<td>Erasure Code</td>
<td>The erasure of code is enabled or disabled for the Nutanix container.</td>
</tr>
<tr>
<td>Size</td>
<td>The Nutanix container size calculated dynamically during discovery.</td>
</tr>
<tr>
<td>Nutanix Virtual Machine Instance</td>
<td>A virtual machine that runs in the Nutanix infrastructure.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix Controller VM.</td>
</tr>
<tr>
<td>VM ID</td>
<td>The universally unique identifier (UUID) of the Nutanix Controller VM.</td>
</tr>
<tr>
<td>Object ID</td>
<td>The ID of the Nutanix Controller VM.</td>
</tr>
<tr>
<td>CPU Count</td>
<td>The number of CPUs that the VM has.</td>
</tr>
<tr>
<td>Memory</td>
<td>The memory size of the Nutanix Controller VM in Megabytes (MB).</td>
</tr>
<tr>
<td>State</td>
<td>The power state of the Nutanix Controller VM (on/off).</td>
</tr>
<tr>
<td>Description</td>
<td>The description defined for the virtual machine in the Nutanix console: Prism Central or Prism Element.</td>
</tr>
<tr>
<td>Nutanix Prism Central</td>
<td>Multi-cluster manager responsible for managing multiple Acropolis Clusters to provide a single, centralized management interface.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the Nutanix Prism Central multi-cluster manager. This attribute uses one of the following conventions:</td>
</tr>
<tr>
<td></td>
<td>· PrismCentral@&lt;IP_ADDRESS&gt;</td>
</tr>
<tr>
<td></td>
<td>· PrismCentral@&lt;DNS_NAME&gt;</td>
</tr>
<tr>
<td>Object_id</td>
<td>The ID of the Nutanix Prism Central. This attribute has the same value as the Name attribute, using one of the following conventions:</td>
</tr>
<tr>
<td></td>
<td>· PrismCentral@&lt;IP_ADDRESS&gt;</td>
</tr>
<tr>
<td></td>
<td>· PrismCentral@&lt;DNS_NAME&gt;</td>
</tr>
<tr>
<td>Key Value (cmdb_key_value) - If your organization uses tags to distinguish or identify CIs, the pattern discovers CI tags for Nutanix hosts and virtual machines.</td>
<td></td>
</tr>
<tr>
<td>Configuration item</td>
<td>The reference to the CI for which the tag is defined.</td>
</tr>
<tr>
<td>key</td>
<td>The category key.</td>
</tr>
<tr>
<td>value</td>
<td>The category value.</td>
</tr>
<tr>
<td>object_id</td>
<td>The Object ID for the Nutanix host or Nutanix virtual machine.</td>
</tr>
</tbody>
</table>

The graphic illustrates CIs that are part of Acropolis discovery in deployments that use Nutanix AHV for virtualization.
The graphic illustrates CIs that are part of Acropolis discovery in deployments that use VMware software for virtualization.
CI relationships

These relationships are created to support Nutanix discovery:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nutanix Cluster (cmdb_ci_nutanix_cluster)</td>
<td>Define resources for::Get resources from Nutanix Storage Pool (cmdb_ci_nutanix_storage_pool)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Members::Member of</td>
<td>Nutanix Host (cmdb_ci_nutanix_host)</td>
</tr>
<tr>
<td>Nutanix Virtual Machine Instance</td>
<td>Registered::Has registered</td>
<td>Nutanix Host (cmdb_ci_nutanix_host)</td>
</tr>
<tr>
<td>(cmdb_ci_nutanix_vm_instance)</td>
<td>Hosted on::Hosts</td>
<td>Nutanix Cluster (cmdb_ci_nutanix_cluster)</td>
</tr>
<tr>
<td>Nutanix Controller VM</td>
<td>Runs on::Runs</td>
<td>Nutanix Virtual Machine Instance</td>
</tr>
<tr>
<td>(cmdb_ci_nutanix_controller_vm)</td>
<td></td>
<td>(cmdb_ci_nutanix_vm_instance)</td>
</tr>
<tr>
<td></td>
<td>Runs on::Runs</td>
<td>Nutanix Virtual Machine Instance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(cmdb_ci_vmware_instance)</td>
</tr>
<tr>
<td></td>
<td>Hosted on::Hosts</td>
<td>Nutanix Prism Central</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(cmdb_ci_nutanix_prism_central)</td>
</tr>
<tr>
<td>Nutanix Storage Pool</td>
<td>Contains::Contained by</td>
<td>Nutanix Storage Container</td>
</tr>
<tr>
<td>(cmdb_ci_nutanix_storage_pool)</td>
<td></td>
<td>(cmdb_ci_nutanix_storage_container)</td>
</tr>
<tr>
<td>CI</td>
<td>Relationship</td>
<td>CI</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Hosted on::Hosts</td>
<td>Nutanix Prism Central</td>
</tr>
<tr>
<td>cmdb_ci_nutanix_prism_central</td>
<td></td>
<td>(cmdb_ci_nutanix_prism_central)</td>
</tr>
<tr>
<td>VMware Cluster</td>
<td>Define resources for::Get resources from</td>
<td>Nutanix Storage Pool</td>
</tr>
<tr>
<td>cmdb_ci_vmware_cluster</td>
<td></td>
<td>(cmdb_ci_nutanix_storage_pool)</td>
</tr>
</tbody>
</table>

**Dashboard**

The dashboard contains reports that query the CMDB and display the results. The information is displayed only after the pattern has discovered the infrastructure.

To display the dashboard, navigate to **Self-Service > Dashboards > All > Nutanix**.
The reports on the dashboard provide the following information. They are listed in the table by the numbers shown in the graphic.

<table>
<thead>
<tr>
<th>Report</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Nutanix Clusters</td>
<td>Number of Nutanix clusters found.</td>
</tr>
<tr>
<td>2 Nutanix Hosts</td>
<td>Number of Nutanix hosts found.</td>
</tr>
<tr>
<td>3 Nutanix Storage Containers</td>
<td>Number of Nutanix storage containers found.</td>
</tr>
<tr>
<td>4 Nutanix Storage Container Distro</td>
<td>The distribution of Nutanix storage containers free space and capacity, in bytes.</td>
</tr>
<tr>
<td>5 Nutanix Storage Pool Capacity</td>
<td>The capacity of the Nutanix storage pool, in bytes.</td>
</tr>
<tr>
<td>6 Nutanix VM Summary</td>
<td>The Nutanix VMs grouped by state: On, off, terminated.</td>
</tr>
<tr>
<td>7 Nutanix Controller VMs</td>
<td>The Nutanix Controller VMs grouped by hypervisor type.</td>
</tr>
</tbody>
</table>

**Troubleshooting**

If the mapping result is not as you expected, perform basic troubleshooting.

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery fails. The discovery message contains the information about an error caused by the REST timeout.</td>
<td>There are many CIs sending the REST call response in the deployment. The MID Server cannot process the REST call response without exceeding the time limit controlled by the mid.sa.cloud.request_timeout parameter.</td>
<td>By default, the mid.sa.cloud.request_timeout parameter is set to 30000 milliseconds. Increase the value of this parameter on the relevant MID Server and run discovery again.</td>
</tr>
<tr>
<td>Pattern designer fails during a debug session. The pattern designer message contains information about an error caused by a timeout.</td>
<td>The pattern designer fails because of a timeout during pattern debugging (and not during discovery).</td>
<td>By default, the sa.debugger.max_timeout parameter is set to 240 seconds. Increase the value of this parameter on the relevant MID Server.</td>
</tr>
</tbody>
</table>

**Oracle Database 12c discovery**

ServiceNow® Discovery can discover pluggable databases (PDBs) and container databases (CDBs), which are features of Oracle Database 12c.
Request apps on the Store

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store. For cumulative release notes information for all released apps, refer to the ServiceNow Store version history release notes.

About this pattern

A PDB is a portable collection of schemas, schema objects, and non-schema objects that appear to an Oracle Net client as a separate database. A collection of PDBs is called a container database (CDB).

For more information about PDBs and CDBs, see What are CDBs, PDBs and Containers?

Prerequisites

Set the user credentials and grant permissions:

- For Windows, see Windows credentials
- For Unix, see SSH credentials
- For the Oracle database, see Oracle database discovery

Set the Applicative credentials. See Applicative credentials

Allow the user to run queries on the following tables:

- V$DATAFILE
- V$TEMPFILE
- V$LOG
- DBA_FREE_SPACE
- V$DATABASE
- V$PDBS

Data collected during Discovery

Discovery collects the following data:

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_db_ora_pdb_instance: Relation Type : Managed By</td>
<td></td>
</tr>
<tr>
<td>sid</td>
<td>Name of the PDB instance</td>
</tr>
<tr>
<td>name</td>
<td>Name of the CI in CMDB. This is a combination of sid and the hostname.</td>
</tr>
<tr>
<td>version</td>
<td>Version of the PDB as taken from the CDB instance</td>
</tr>
<tr>
<td>edition</td>
<td>Edition of the PDB as taken from the CDB instance</td>
</tr>
<tr>
<td>install_directory</td>
<td>Install directory of the PDB as taken from the CDB instance</td>
</tr>
<tr>
<td>oracle_home</td>
<td>Oracle home of the PDB as taken from the CDB instance</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>cdb_name</td>
<td>Name of the cdb sid that contains the PDB. This refers to the Oracle database instance configuration item (CI)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>cmdadb_ci_db_instance_size: Relation Type : Contained By</th>
</tr>
</thead>
<tbody>
<tr>
<td>sid</td>
</tr>
<tr>
<td>name</td>
</tr>
<tr>
<td>db_size</td>
</tr>
<tr>
<td>used_size</td>
</tr>
<tr>
<td>free_size</td>
</tr>
</tbody>
</table>

The following is a dependency view of the relationship between PDBs, CDBs, and containers.
Oracle GoldenGate discovery

Discovery and Service Mapping find Oracle GoldenGate version 12c components using the Oracle GoldenGate pattern.

You can use this pattern on the Now Platform using Kingston, London, or Madrid.

Discovery uses the Oracle GoldenGate pattern to perform horizontal discovery to collect data into the $report_file file. Service Mapping performs top-down discovery on the $report_file file to find outgoing Oracle GoldenGate connections.

Prerequisites

Credentials

Configure the following credentials, depending on what type of host machine that the Oracle GoldenGate Server is installed on:

- Windows credentials
- SSH credentials for UNIX (IBM AIX, Hewlett-Packard HP-UX, or Oracle Solaris) and Linux operating systems

User access

Give the UNIX OS user permissions to read the $report_file. The $report_file is the report file that is extracted from the Oracle GoldenGate manager/replicat/extract process, with the extension .rpt. For example:

Configuration file:

```
/base/ggs_beta/oracle/bt01pims/ggs/dirprm/mgr.prm REPORTFILE
```

Report file:

```
/base/ggs_beta/oracle/bt01pims/ggs/dirrpt/MGR.rpt PROCESSID MGR
```

Permissions to run commands

Give the UNIX OS user permissions to run the following commands against the Oracle GoldenGate Server:

```
<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;privilege command&quot; + &quot;is -d &quot; + $rpt_base_dir + &quot;*.rpt</td>
<td>sort &quot;</td>
<td>Mandatory</td>
<td>List all report files under the Oracle GoldenGate report base directory.</td>
</tr>
<tr>
<td>&quot;privilege command&quot; + &quot;is -d &quot; + $prm_base_dir + &quot;*.prm</td>
<td>sort &quot;</td>
<td>Mandatory</td>
<td>List all parameter files under the Oracle GoldenGate parameter base directory.</td>
</tr>
</tbody>
</table>
```

Give the Windows OS user permissions to run the following commands against the Oracle GoldenGate Server:
## Retrieve data

Retrieve data by parsing:

- The Oracle GoldenGate manager report file (report_file) to retrieve related configuration item (CI) names and counter information.
- Variables in the process command line to retrieve the install folder, manager process name, configuration file, and parameters file.
- Variables in the Oracle GoldenGate installation folder to retrieve a list of parameter and configuration files.

### EVAL closure functions

Use the following EVAL closure functions to remove duplicate entries from the extract process and the replicat process.

- var tableWithoutDuplicates = ";tableWithoutDuplicates = DuplicateRemover.removeDuplicates(${extracts},("name");CTX.setAttribute("extracts", tableWithoutDuplicates);
- var tableWithoutDuplicates = ";tableWithoutDuplicates = DuplicateRemover.removeDuplicates(${replicats},("name");CTX.setAttribute("replicats", tableWithoutDuplicates);

Use the following EVAL closure functions to count the number of extract processes and replicat processes.

- return ${cmdb_ci_appl_ora_gg_replicat(*).config_file}.size()
- return ${cmdb_ci_appl_ora_gg_extract(*).config_file}.size()

Use the following EVAL closure function to return the privileged command.

- return $(ctx).getDiscoveryProvider(com.snc.sw.dto.ProviderType.SSH).getPrivilegedCommand();

Use the following EVAL closure function to return the discovery type.

- ctx.getWork().getDiscoveryType();

Use the following EVAL closure function to extract the version from the installation directory path if it is empty.

- inst_dir = ${install_directory}if(inst_dir.isEmpty()){return ${version}}if(inst_dir.startsWith(©/©)){return inst_dir.replaceAll(©/.*/©,'')}if(!inst_dir.startsWith(©/©) && !inst_dir.isEmpty()){return inst_dir.replaceAll(©.*\\©,'')}  

### Applicative credentials

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
1. Navigate to Discovery > Credentials.
2. Click New.
3. Click Applicative Credentials.
4. On the form, fill in the fields.

**Applicative Credentials form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Credential name. An example is oracle_db_user.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this credential for discovery. Select this check box to enable discovery.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Credentials that you may or may not want to apply to All MID servers in your network, or to one or more Specific MID servers. Select Specific MID servers.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>MID Servers that the credentials apply to. Select the required MID Server. This field appears when Specific MID servers is selected from the Applies to field.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which the platform tries this credential as it attempts to log on to devices. A smaller number indicates that the credential appears higher in the list. Establish the credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number, or none, the instance tries the credentials in a random order. The default value is 100.</td>
</tr>
<tr>
<td>User name</td>
<td>Name of the user of this applicative credential. An example is oracle_db_user. You can use any user for the credential for this pattern, because the information is extracted from a local cache.</td>
</tr>
<tr>
<td>Password</td>
<td>Not required. You can leave this field blank or enter any value.</td>
</tr>
<tr>
<td>CI type</td>
<td>CI type for which this credential is used: Storage Server (cmdb_ci_storage_server).</td>
</tr>
</tbody>
</table>

**Note:** ServiceNow applications refer to devices and applications that comprise an application service as configuration items (CIs).

5. Click Submit.

**Entry point**

For top-down discovery, use the Oracle GoldenGate IP address and specify the MID Server.
Data collected by Discovery during horizontal discovery

Discovery uses the Oracle GoldenGate pattern to collect the data described in the following table.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI cmdb_ci_appl_oracle_golden_gate</td>
<td>Name (name) Name of the CI in the CMDB (&lt;process name&gt;@&lt;source db&gt;).</td>
</tr>
<tr>
<td></td>
<td>Version (version) Version of the Oracle GoldenGate installation.</td>
</tr>
<tr>
<td></td>
<td>Source DB (source_db) Manager process source database system identifier (SID).</td>
</tr>
<tr>
<td></td>
<td>Replicats count (count_replicat) Counter of replicat processes that are managed by the Oracle GoldenGate manager instance.</td>
</tr>
<tr>
<td></td>
<td>Extracts count (count_extract) Counter of extract processes that are managed by the Oracle GoldenGate manager instance.</td>
</tr>
<tr>
<td></td>
<td>Configuration file (config_file) Parameter file of the Oracle GoldenGate process. Specify the path to the configuration file and the file name, &lt;name&gt;.prm</td>
</tr>
<tr>
<td></td>
<td>Replicat File (report_file) Report file of the Oracle GoldenGate replicat process. Specify the path to the report file and the file name, &lt;name&gt;.rpt</td>
</tr>
<tr>
<td></td>
<td>Type (type) Type of the Oracle GoldenGate installation. Specify the Oracle GoldenGate for Oracle technologies.</td>
</tr>
<tr>
<td></td>
<td>Installation directory (install_directory) Folder that contains all the Oracle GoldenGate setup, configuration, libraries, and executable files.</td>
</tr>
<tr>
<td>Related CI cmdb_ci_appl_ora_gg_replicat</td>
<td>Name (name) Name of the CI in the CMDB (&lt;process name&gt;@&lt;source db&gt;).</td>
</tr>
<tr>
<td></td>
<td>Replicat file (report_file) Replicat process report file. Specify the path to the report file and the file name &lt;name&gt;.rpt</td>
</tr>
<tr>
<td></td>
<td>Configuration file (config_file) Parameter file of the replicat process. Specify the path to the configuration file and the file name &lt;name&gt;.prm</td>
</tr>
<tr>
<td></td>
<td>Installation directory (install_directory) Folder that contains all the Oracle GoldenGate setup, configuration, libraries, and executable files.</td>
</tr>
<tr>
<td></td>
<td>Version (version) Version of the Oracle GoldenGate installation.</td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Source DB (source_db)</td>
<td>Manager process source database SID.</td>
</tr>
<tr>
<td>Operational status (operational_status)</td>
<td>Operational status of the CI. Select <strong>Operational</strong>.</td>
</tr>
<tr>
<td>Related CI cmdb_ci_appl_oracle_golden_gate</td>
<td>Name of the CI in the CMDB (&lt;process name&gt;@&lt;source db&gt;).</td>
</tr>
<tr>
<td>Configuration file (config_file)</td>
<td>Parameter file of the extract process.</td>
</tr>
<tr>
<td>Version (version)</td>
<td>Version of the Oracle GoldenGate installation.</td>
</tr>
<tr>
<td>Installation directory (install_directory)</td>
<td>Folder that contains all the Oracle GoldenGate setup, configuration, libraries, and executable files.</td>
</tr>
<tr>
<td>Source DB (source_db)</td>
<td>Manager process source database SID.</td>
</tr>
<tr>
<td>Operation status (operational_status)</td>
<td>Operational status of the CI. Select <strong>Operational</strong>.</td>
</tr>
</tbody>
</table>

**CI relationships**

These relationships are created to support Oracle GoldenGate discovery.

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main cmdb_ci_appl_oracle_golden_gate</td>
<td>Manages::Managed by</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>cmdb_ci_appl_oracle_golden_gate</td>
<td>Managed by::Manages</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>cmdb_ci_appl_oracle_golden_gate</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>Related CI cmdb_ci_appl_oracle_golden_gate</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>cmdb_ci_appl_oracle_golden_gate</td>
<td>Managed by::Manages</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>cmdb_ci_appl_oracle_golden_gate</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>Related CI cmdb_ci_appl_oracle_golden_gate</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>cmdb_ci_appl_oracle_golden_gate</td>
<td>Managed by::Manages</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
<tr>
<td>cmdb_ci_appl_oracle_golden_gate</td>
<td>Runs on::Runs</td>
<td>cmdb_ci_appl_oracle_golden_gate</td>
</tr>
</tbody>
</table>
Data collected by Service Mapping during top-down discovery

To discover the Oracle GoldenGate process, use the TCP entry point with the proper host and port of the Oracle GoldenGate process.

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI cmdb_ci_appl_oracle_golden_gate</td>
<td>Name of the CI in the CMDB (&lt;process name&gt;@&lt;source db&gt;).</td>
</tr>
<tr>
<td>Name (name)</td>
<td>Version of the Oracle GoldenGate installation.</td>
</tr>
<tr>
<td>Version (version)</td>
<td>Installation directory (install_directory) Folder that contains all the Oracle GoldenGate setup, configuration, libraries, and executable files.</td>
</tr>
<tr>
<td>Installation directory (install_directory)</td>
<td>Configuration file (config_file) Parameter file of the Oracle GoldenGate process.</td>
</tr>
<tr>
<td>Replicat file (report_file)</td>
<td>Source database (source_db) Manager process source database SID.</td>
</tr>
<tr>
<td>Source database (source_db)</td>
<td>Extract process count (count_extract) Counter of extract processes that are managed by the Oracle GoldenGate manager instance.</td>
</tr>
<tr>
<td>Extract process count (count_extract)</td>
<td>Replicat process count (count_replicat) Counter of replicat processes that are managed by the Oracle GoldenGate manager instance.</td>
</tr>
</tbody>
</table>

Pivotal Cloud Foundry discovery

Discovery finds Pivotal Cloud Foundry (PCF) version 2.4 components using the Pivotal Cloud Foundry pattern.

You can use this pattern on the Now Platform using Kingston, London, or Madrid.

Prerequisites

Cloud Foundry credentials

Create the Cloud Foundry credentials.

1. Navigate to Discovery > Credentials.
2. Click New.
3. Click CloudFoundry Credentials.
4. On the form, fill in the fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Credential name. As an example, pcf_user credentials</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this credential for discovery. Select this check box to enable discovery.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Credentials that you may or may not want to apply to All MID servers in your network, or to one or more Specific MID servers. Select Specific MID servers.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which the platform tries this credential as it attempts to log on to devices. A smaller number indicates that the credential appears higher in the list. Establish the credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number, or none, the instance tries the credentials in a random order. The default value is 100.</td>
</tr>
<tr>
<td>User name</td>
<td>Name of the user of these credentials. Avoid leading or trailing spaces in user names. A warning appears if the platform detects leading or trailing spaces in the user name. An example is pcf_user. Use the client_id that is used to generate the JWT token.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for the user of these credentials. Use the client_secret that is used to generate the JWT token.</td>
</tr>
<tr>
<td>Credential alias</td>
<td>Alternate name for these credentials. As an example, cf If this field is closed, click the unlock icon to unlock the field.</td>
</tr>
</tbody>
</table>

5. Click Update.

### JSON web token credentials

To verify the JSON web token (JWT), perform the following token validation steps.

The format of the JWT token is: [https://docs.cloudfoundry.org/api/uaa/version/4.27.0/index.html#jwt-bearer-token-grant](https://docs.cloudfoundry.org/api/uaa/version/4.27.0/index.html#jwt-bearer-token-grant)

1. Run the following cURL script against the User Account and Authentication (UAA) server API and verify the results:

   ```bash
   ```

   - If this command does not generate a token, verify that the credentials in the ServiceNow repository are correct, otherwise go to step 2.

2. Run the following cURL script against Pivotal Cloud Foundry API:

   ```bash
   ```
If this command does not retrieve the PCF spaces information, verify the credentials on the ServiceNow credentials repository.

**Credentials for V2 API elements**

Provide a read only user with permission to use the following API elements:

- `~/v2/organizations`
- `~/v2/info`
- `~/v2/spaces`
- `~/v2/domains`
- `~/v2/routes`
- `~/v2/quota_definitions`
- `~/v2/space_quota_definitions`
- `~/v2/apps`

**Create a serverless Discovery schedule**

1. Navigate to **Discovery > Discovery Schedules**.
2. Click **New**.
3. On the form, fill in the fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this Discovery schedule, for example, pcf</td>
</tr>
<tr>
<td>Discover</td>
<td>For the Discover type, select <strong>Serverless</strong></td>
</tr>
<tr>
<td>MID Server</td>
<td>MID Servers that the credentials apply to. Specify the required MID Server, for example, Discovery_Server</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling this credential for discovery. Select this check box to enable discovery.</td>
</tr>
<tr>
<td>Credential alias</td>
<td>Alternate name for these credentials. Use the alias associated with the Pivotal Cloud Foundry credentials you created earlier.</td>
</tr>
</tbody>
</table>

4. Right-click the header of the Discovery Schedule form and select **Save**.
5. Click the Serverless Execution Patterns tab.
6. Click New.
7. On the form, fill in the fields:

   **Serverless Execution Pattern form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this Serverless Execution Pattern, for example, pcf1</td>
</tr>
<tr>
<td>Pattern</td>
<td>Select CloudFoundry</td>
</tr>
<tr>
<td>Proxy Host</td>
<td>Fully qualified domain name of the machine on which you are installing the proxy server. Specify Global.</td>
</tr>
<tr>
<td>Active</td>
<td>Check box to enable this schedule for discovery. Select this check box.</td>
</tr>
</tbody>
</table>

8. Right-click the header of the Serverless Execution Patterns form and select Save.
9. Click the Serverless Execution Patterns tab
10. Create and define the serverless execution pattern as described in the product documentation.
    Configure the parameters required by the Cloud Foundry pattern as follows:

   **Discovery Pattern Launcher parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization</td>
<td>Name of the organization that is required to be discovered, or all where the input is &quot;*&quot;.</td>
</tr>
<tr>
<td>CredentialAlias</td>
<td>Alternate name for these credentials. Use the alias associated with the Cloud Foundry credentials you created earlier.</td>
</tr>
<tr>
<td>URL_CF</td>
<td>API URL of the Cloud Controller that is used to retrieve the information from Cloud Foundry.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>URL_CF_AUTH</td>
<td>UAA server API that is used to generate the token. Enter the value in the format: <a href="https://uaa.sys.dev.phx.pcf.example.com/oauth/token">https://uaa.sys.dev.phx.pcf.example.com/oauth/token</a></td>
</tr>
</tbody>
</table>

### 11. Update

To start discovery, navigate to the Serverless Discovery definition and click **Discover Now**.

### EVAL closure functions

Use the following EVAL closure functions to concatenate all organizations with “,” as a separator.

```javascript
var rtrn = ''; 
var organizations = ${organizations[*].name};
var beforeReturn = ''; 
var separator = ''; 
for (var i = 0; i < organizations.size(); i++)
{
    beforeReturn += separator + organizations.get(i);
    separator = ',';
} 
rtrn = beforeReturn;
```

Use the following EVAL closure functions to return 1 if the operational status is active and 2 if it is not active.

```javascript
var rtrn = ''; 
var currentOrgStatus = ${organizations[*].operational_status};
var status = currentOrgStatus.toLowerCase();
if(status == "active")
{
    rtrn = "1";
}
else {
    rtrn = "2";
}
```

Use the following EVAL closure function to return 1 if the operational status is started and 2 if it is not started.

```javascript
var rtrn = ''; 
var currentAppStatus = ${apps[*].state};
var status = currentAppStatus.toLowerCase();
if(status == "started")
{
    rtrn = "1";
}
else {
    rtrn = "2";
}
```

### CloudFoundry Get Call

The CloudFoundry Get Call custom operation handles the generation of tokens. It uses the credentials and URL_CF_AUTH context variable to generate a token that is used to create API calls.
When a token is generated, the URL_CF is used to create the API calls defined in the **Resource** field. The custom operation also handles pagination where the response has more than one page.

CloudFoundry Get Call parses the input JSON to populate the tables and variables.

**Data collected by Discovery during horizontal discovery**

**Entry point**

For top-down discovery, use the Cloud Foundry IP address of the Cloud Foundry host and specify the MID Server.

**Collected information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main CI: cmdb_ci_pcf_foundation</td>
<td>Cloud Foundry cluster entry name</td>
</tr>
<tr>
<td>Name</td>
<td>Cloud Foundry cluster entry name</td>
</tr>
<tr>
<td>ip_address</td>
<td>Cloud Foundry host IP.</td>
</tr>
<tr>
<td>port</td>
<td>Cloud Foundry service port.</td>
</tr>
<tr>
<td>short_description</td>
<td>Cloud Foundry service build number.</td>
</tr>
<tr>
<td>Organisation</td>
<td>Organization list, managed by PCF cluster. The entries are separated using a &quot;,&quot; comma separator.</td>
</tr>
</tbody>
</table>

Related CI: cmdb_ci_pcf_organization
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Organization name</td>
</tr>
<tr>
<td>URL</td>
<td>Organization unique URL</td>
</tr>
<tr>
<td>pcf_guid</td>
<td>Organization UID in CF</td>
</tr>
<tr>
<td>operational_status</td>
<td>Organization status</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_space</td>
<td>Space name</td>
</tr>
<tr>
<td>Name</td>
<td>Space unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Space UID in CF</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_appl</td>
<td>Application name</td>
</tr>
<tr>
<td>Name</td>
<td>Application unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Application UID in CF</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_route</td>
<td>Route name</td>
</tr>
<tr>
<td>Name</td>
<td>Route unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Route UID in CF</td>
</tr>
<tr>
<td>port</td>
<td>Route port</td>
</tr>
<tr>
<td>type</td>
<td>Route type</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_quota</td>
<td>Quota name</td>
</tr>
<tr>
<td>Name</td>
<td>Quota unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Quota UID in CF</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_domain</td>
<td>Domain name</td>
</tr>
<tr>
<td>Name</td>
<td>Domain unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Domain UID in CF</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_space_instance</td>
<td>Space service instance name</td>
</tr>
<tr>
<td>Name</td>
<td>Space service instance unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Space service instance UID in CF</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_service_plan</td>
<td>Service plan instance name</td>
</tr>
<tr>
<td>Name</td>
<td>Service plan unique URL</td>
</tr>
<tr>
<td>URL</td>
<td>Space service instance UID in CF</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_service</td>
<td>Service name</td>
</tr>
<tr>
<td>Name</td>
<td>Service unique URL</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------</td>
</tr>
<tr>
<td>pcf_guid</td>
<td>Service UID in CF</td>
</tr>
</tbody>
</table>

**CI relationships**

These relationships are created to support Pivotal Cloud Foundry discovery.

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Related CI: cmdb_ci_pcf_space</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_organization</td>
</tr>
<tr>
<td>cmdb_ci_pcf_space</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_appl</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>cmdb_ci_pcf_appl</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_route</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>cmdb_ci_pcf_route</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_quota</td>
<td>Uses::Used by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>cmdb_ci_pcf_quota</td>
<td>Uses::Used by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_domain</td>
<td>Uses::Used by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>cmdb_ci_pcf_domain</td>
<td>Owns::Owned by</td>
<td>cmdb_ci_pcf_organization</td>
</tr>
<tr>
<td></td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_space_instance</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>cmdb_ci_pcf_space_instance</td>
<td>Contains::Contained by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>Related CI: cmdb_ci_pcf_service_plan</td>
<td>Uses::Used by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
<tr>
<td>cmdb_ci_pcf_service_plan</td>
<td>Uses::Used by</td>
<td>cmdb_ci_pcf_space</td>
</tr>
</tbody>
</table>

**Pure Storage FlashBlade discovery**

Discovery uses the FlashBlade Pure Storage pattern to find FlashBlade components.

Discovery can find Pure Storage FlashBlade version 4.

You can use this pattern on the Now Platform using Jakarta Patch 9, Kingston Patch 5, London, or Madrid.

**Prerequisites**

**SSH credentials**

On the Now Platform®, configure a user for accessing the Unix OS.

**Applicative credentials**
On the Now Platform®, configure applicative credentials as follows:

- Define the OS use with permissions to run the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pureman</td>
<td>-</td>
<td>Mandatory</td>
<td>Accesses the list of pure storage commands and uses them for classification.</td>
</tr>
<tr>
<td>Echo</td>
<td>$username$</td>
<td>Mandatory</td>
<td>Fills in the username for the temporary variable.</td>
</tr>
<tr>
<td>&quot;pureadmin list&quot;</td>
<td>+ $usr + ' --api-token --expose --csv&quot;</td>
<td>Mandatory</td>
<td>Gets the user API token to generate the x-auth for the REST call commands.</td>
</tr>
</tbody>
</table>

This OS user must also have permissions to run the following REST API calls:

- `/api/login` – Gets the x-auth token reader from the response in Groovy.
- `/api/1.2/file-systems` – Retrieves information about the file systems.
- `/api/1.2/blades` – Retrieves information about blades.
- `/api/1.2/dns` – Retrieves information about DNS.
- `/api/1.2/hardware` – Retrieves information about the storage hardware.
- `/api/1.4/network-interfaces` – Retrieves information about network interfaces.

- Enter any value for the password or leave it empty. This password is not relevant.

**Access to the port used for REST calls**

On relevant MID Servers, provide a user with access to the port used for REST calls.

**Data collected by Discovery during horizontal discovery**

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Server (cmdb_ci_storage_server)</td>
<td>The attributes of the storage server provided by FileShare for PureBlade.</td>
</tr>
<tr>
<td>IP address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Serial number (serial_number)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Pure Storage Processor (cmdb_ci_storage_processor_pure)</td>
<td>The attributes of the PureBlade storage processor.</td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Serial number (serial_number)</td>
<td></td>
</tr>
<tr>
<td>Device ID (device_id)</td>
<td></td>
</tr>
<tr>
<td>Model number (model_number)</td>
<td></td>
</tr>
<tr>
<td>Serial (serial)</td>
<td></td>
</tr>
<tr>
<td>Slot (slot)</td>
<td></td>
</tr>
<tr>
<td>Type (type)</td>
<td></td>
</tr>
<tr>
<td>Table and field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Status (status)</td>
<td>Pure Storage File Share (cmdb_ci_storage_fileshare_pure)</td>
</tr>
<tr>
<td></td>
<td>Name (name)</td>
</tr>
<tr>
<td></td>
<td>Path (path)</td>
</tr>
<tr>
<td></td>
<td>Protocols (protocols)</td>
</tr>
<tr>
<td></td>
<td>Provisioned Space (provisioned_space)</td>
</tr>
<tr>
<td></td>
<td>Unique Space (unique_space)</td>
</tr>
<tr>
<td></td>
<td>Snapshots Space (snapshots_space)</td>
</tr>
<tr>
<td></td>
<td>Total Physical Space (total_physical_space)</td>
</tr>
<tr>
<td></td>
<td>Shared space (shared_space)</td>
</tr>
<tr>
<td></td>
<td>Data Reduction (data_reduction)</td>
</tr>
<tr>
<td></td>
<td>System Space (system_space)</td>
</tr>
<tr>
<td></td>
<td>Storage Device (cmdb_ci_storage_device)</td>
</tr>
<tr>
<td></td>
<td>Device ID (device_id)</td>
</tr>
<tr>
<td></td>
<td>Name (name)</td>
</tr>
<tr>
<td></td>
<td>Serial number (serial_number)</td>
</tr>
<tr>
<td></td>
<td>Model number (model_number)</td>
</tr>
<tr>
<td></td>
<td>Storage type (storage_type)</td>
</tr>
</tbody>
</table>

The attributes of the storage FileShare.

The attributes of the storage device contained by the storage server.

The graphic illustrates CIs that are part of FlashBlade discovery.
CI relationships

The Amazon AWS DynamoDB pattern creates the following CI relationships:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage server (cmdb_ci_storage_server)</td>
<td>Provided by: Provides</td>
<td>Fileshare (cmdb_ci_storage_fileshare_pure)</td>
</tr>
<tr>
<td>Storage server (cmdb_ci_storage_server)</td>
<td>Contains: Contained by</td>
<td>Storage device (cmdb_ci_storage_device)</td>
</tr>
<tr>
<td>Storage server (cmdb_ci_storage_server)</td>
<td>Contains: Contained by</td>
<td>Pure Storage Processor (cmdb_ci_storage_processor_pure)</td>
</tr>
</tbody>
</table>

Red Hat JBoss Fuse discovery

Discovery and Service Mapping can find and map the Fuse application server using the JBoss Fuse pattern.

Prerequisites

Credentials
Configure SSH credentials in the Credentials module of the ServiceNow platform for accessing the server hosting the Fuse server.

**User**

Provide a user with the read rights for accessing the following directories:

- `$install_directory + "/readme.txt"
  
  Access to this directory allows getting version information.
- `/etc/services`
  
  Access to this directory allows reading the hosts service definition.

**Data collected by Discovery during horizontal discovery**

<table>
<thead>
<tr>
<th>Table and Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jboss Fuse (cmdb_ci_appl_jboss_fuse)</td>
<td>The configuration directory of the Fuse server.</td>
</tr>
<tr>
<td>config_directory</td>
<td>The configuration file of the Fuse server.</td>
</tr>
<tr>
<td>config_file</td>
<td>The installation folder of the Fuse server.</td>
</tr>
<tr>
<td>install_folder</td>
<td>The version of the Fuse server.</td>
</tr>
<tr>
<td>version</td>
<td>The name configured for the Fuse server during installation.</td>
</tr>
</tbody>
</table>

**Data collected by Service Mapping during top-down discovery**

Service Mapping discovers the connection between the JBoss Fuse server and the IBM Informix Dynamic server.

**SAP System ID discovery**

An SAP System Identification code (SID) is a unique identification code for an SAP R/3 installation. An SID is a three-character code like D10, PRD, or E56. An R/3 installation consists of a database server plus several application servers. The SAP System ID pattern is a ServiceNow Store app. For details on system requirements, view the application listing on the ServiceNow Store website.

**Get the app on the ServiceNow Store**

Visit the ServiceNow Store website to get the app.

**Prerequisites**

Google Cloud Platform patterns work with Cloud Management. The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt). Use your current Cloud Management instance or request a subscription from your sales representative.

Other prerequisites are the same as for discovery of the SAP application.
Components

Each SAP pattern app (CI, DI, JC, ASCS, SCS, and ERS) creates relations to the SAP SID CI `cmdb_ci_sap_sid` based on its version and SID name. If such a CI doesn't exist in the CMDB, the app also creates the SAP SID CI.

Horizontal discovery creates a new CI for each app. The SID name and version show which SAP app is related to which R/3 installation. The relationship to `cmdb_ci_sap_sid` is shown for each SAP application CI.

CI relationships

Connections discovered by Service Mapping during the top-down discovery

The CI of Storage Server should be discovered: `cmdb_ci_sap_sid`. Populated attributes: Name - SAP SID name Version – SAP SID version
Example

![Diagram of SAP systems](image)

Veritas Cluster Server

Discovery uses the Unix Cluster – VERITAS Cluster pattern to find Veritas Cluster Server components. You can use this pattern on the Now Platform using Kingston Patch 8, London, or Madrid.

Prerequisites

**SSH credentials**

On the Now Platform®, configure the SSH credentials for a user with elevated rights for running the `cat` command. For more information about commands, see Service Mapping commands requiring a privileged user under Service Mapping documentation.

**Permissions to read the Veritas Cluster Server configuration file**

Give the Unix OS user permissions to read the following configuration file: `/etc/VRTSvcs/conf/config/main.cf`.

**Permissions to run Veritas Cluster commands**

Give the Unix OS user permissions to run the following commands against the Veritas Cluster Server:

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;privilege user&quot; + <code>cat /etc/VRTSvcs/conf/config/main.cf</code></td>
<td>-</td>
<td>Mandatory</td>
<td>Read the Veritas Cluster Server configuration file.</td>
</tr>
<tr>
<td>Command</td>
<td>Parameter</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/haclus -value ClusterUUID 2&gt;/dev/null</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the Veritas Cluster Server cluster UUID.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/haclus -value ClusterName 2&gt;/dev/null</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the Veritas Cluster Server cluster name.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/haclus -value EngineVersion</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the Veritas Cluster Server cluster version.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/haclus -value ClusterAddress</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the Veritas Cluster Server cluster address.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/haclus -value ClusState</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the Veritas Cluster Server cluster status.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/hares -state</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the cluster resources of the Veritas Cluster Server.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/hares -display</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the cluster resource type of the Veritas Cluster Server.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/hares -display</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the cluster resource group of the Veritas Cluster Server.</td>
</tr>
<tr>
<td>&quot;privilege user&quot; + /opt/VRTSvcs/bin/hagrp -state</td>
<td>-</td>
<td>Mandatory</td>
<td>Get the resource group name and status of the Veritas Cluster Server.</td>
</tr>
</tbody>
</table>

**Classification probe for triggering the UNIX Cluster – VERITAS Cluster pattern**

Verify that the classification probe is set to trigger the UNIX Cluster – VERITAS Cluster pattern:

1. Navigate to **Discovery Definition > CI Classification > CI Classification > UNIX**.
2. In the **UNIX Classification** list, click **Solaris** or **Linux**.
3. On the **Triggers probes** tab, check that the **HorizontalDiscoveryProbe-HorizontalPatt** probe is assigned to the **UNIX Cluster – VERITAS Cluster** pattern.
Note: The discovery log shows the error for OS discovery step even if the discovery finished successfully.

System property for the new host class

Add a system property (sys_property) `sa.host_classes` and set the value to `cmdb_ci_unix_cluster`. Creating a new host class for Veritas Cluster servers helps to identify this type of hosts correctly.
Limitations

You cannot customize the Unix Cluster – VERITAS Cluster pattern in the Debug mode in the Pattern Designer.

Data collected by Discovery during horizontal discovery

<table>
<thead>
<tr>
<th>Table and field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unix Cluster (cmdb_ci_unix_cluster)</td>
<td>The attributes of the Unix Cluster.</td>
</tr>
<tr>
<td>IP address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Cluster type (cluster_type)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Cluster version (cluster_version)</td>
<td></td>
</tr>
<tr>
<td>Cluster status (cluster_status)</td>
<td></td>
</tr>
<tr>
<td>Unix Cluster Node (cmdb_ci_unix_cluster_node)</td>
<td>The attributes of the Unix Cluster Node.</td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Cluster (cluster)</td>
<td></td>
</tr>
<tr>
<td>Server (server)</td>
<td></td>
</tr>
<tr>
<td>Node state (node_status)</td>
<td></td>
</tr>
<tr>
<td>IP address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Node (node)</td>
<td></td>
</tr>
<tr>
<td>Server (server)</td>
<td></td>
</tr>
<tr>
<td>Cluster (cluster)</td>
<td></td>
</tr>
<tr>
<td>Resource group status (resource_group_status)</td>
<td></td>
</tr>
<tr>
<td>UNIX Cluster Resource (cmdb_ci_unix_cluster_resource)</td>
<td>The attributes of the UNIX Cluster Resource.</td>
</tr>
<tr>
<td>Resource type (resource_type)</td>
<td></td>
</tr>
<tr>
<td>Name (name)</td>
<td></td>
</tr>
<tr>
<td>Resource status (resource_status)</td>
<td></td>
</tr>
<tr>
<td>Cluster Virtual IPs (cmdb_ci_cluster_vip)</td>
<td>The attributes of the Cluster Virtual IP addresses.</td>
</tr>
<tr>
<td>IP address (ip_address)</td>
<td></td>
</tr>
<tr>
<td>name (name)</td>
<td></td>
</tr>
<tr>
<td>Cluster (cluster)</td>
<td></td>
</tr>
<tr>
<td>Cluster status (cluster_status)</td>
<td></td>
</tr>
<tr>
<td>Node (node)</td>
<td></td>
</tr>
</tbody>
</table>

The graphic illustrates CIs that are part of Veritas Cluster Server discovery.
CI relationships

The Unix Cluster – VERITAS Cluster pattern creates the following CI relationships:

<table>
<thead>
<tr>
<th>CI</th>
<th>Relationship</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unix Cluster <em>(cmdb_ci_unix_cluster)</em></td>
<td>Hosts: Hosted on</td>
<td>Linux Server <em>(cmdb_ci_linux_server)</em></td>
</tr>
<tr>
<td>Unix Cluster Node <em>(cmdb_ci_unix_cluster_node)</em></td>
<td>Hosts: Hosted on</td>
<td>Linux Server <em>(cmdb_ci_linux_server)</em></td>
</tr>
<tr>
<td>Cluster Virtual IPs <em>(cmdb_ci_cluster_vip)</em></td>
<td>Virtualized by: Virtualized</td>
<td>Unix Cluster <em>(cmdb_ci_unix_cluster)</em></td>
</tr>
<tr>
<td>Cluster Virtual IPs <em>(cmdb_ci_cluster_vip)</em></td>
<td>Uses: Used by</td>
<td>Unix Cluster Node <em>(cmdb_ci_unix_cluster_node)</em></td>
</tr>
</tbody>
</table>
Create CI types for Service Mapping and Discovery

Create CI types (or CI classes) for applications and devices, which Service Mapping and Discovery do not support by default.

Verify that there is no existing CI type for this application or device.

Role required: sm_admin, personalize_dictionary, or admin

All applications and hosts in your organization must have a corresponding configuration item (CI) type, which is necessary for discovering and processing applications and hosts correctly.

A wide range of preconfigured CI types that cover most commonly used applications are available to you. If your organization uses a less known or proprietary application that does not have a corresponding CI type, you must create it.

A CI type is a generic notion that is used by several ServiceNow applications, but there are some attributes that are specific to Service Mapping and Discovery.

Note: Service Mapping user interface refers to CI classes as CI types.

A CI type (or class) contains several important definitions that apply to all CIs belonging to it, such as:

- CI attributes are added as fields to the CMDB tables.
- Identifiers help Service Mapping and Discovery to differentiate between new and existing CIs. For example, if there is an Apache Web Server CI type defined in the CMDB, and Service Mapping and Discovery both discover an Apache Web Server CI, it processes it using identifiers. It then recognizes it as an updated version of the Apache Web Server CI that already exists in the system, not a new Apache Web Server CI.
- Related items, both CIs and non-CIs, serve as a reference and provide additional information about the CI. For example, you can define a serial number, a port and an network ARP table as related items for a hardware server. The system does not use these items for identification. After you configure related items for the CI type, you can use horizontal discovery for CIs of this type together with their related items. For information, see Discover related items together with the main CI.

Note: Non-CIs are items that do not extend the Configuration Item (cmdb_ci) table. Examples of non-CIs are ports, serial numbers, or Network ARP tables.
• There are reconciliation rules that help the ServiceNow platform to consolidate CI attributes received from different applications correctly. These rules are necessary for organizations where more than one application participates in the discovery process. Reconciliation rules define how attributes of the same CI discovered by different discovery sources are merged. For example, Service Mapping discovers the version and home directory attributes of an Apache Web Server CI, while Discovery discovers the version and patch level attributes for the same Apache Web Server CI. The Now Platform applies the reconciliation rule and as a result Service Mapping does not overwrite the attributes found by Discovery.

• CMDB stores CI class in the form of a hierarchy where some CI classes are parents to other CI classes, who automatically derive their parent's attributes in addition to attributes you configure specifically for child CI classes. CI class hierarchy is used widely for configuring CI behavior, relationships, and display. In this example, the Apache Web Server CI is a child of the Web Server CI and derives many attributes from its parent, such as name, version, model ID. By default, child CI classes inherit identifiers from their parent CI classes.

In addition to these CI type definitions, the horizontal discovery process uses a CI classification to define to which CI type a CI belongs. Create a device CI classification if you create a CI type for devices using SNMP and a process CI classification for an application CI type.

1. Navigate to Configuration > CI Class Manager.
2. To use an existing CI type as a parent for the new CI type, click Hierarchy to display the CI Classes list. Click the context menu for the required CI type and select Add Child Class.
3. Create a table to store the CI type attributes:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>CI type name. For example, Apache Web Server.</td>
</tr>
<tr>
<td>Name</td>
<td>The table name. For example, cmdb_ci_apache_web_server. Use a name similar to</td>
</tr>
<tr>
<td></td>
<td>the other CMDB classes (for example, u_cmdb_ci_laptop).</td>
</tr>
</tbody>
</table>
### Field

**Extends table**

The table name of the parent CI type. All CI types are extensions of the Configuration Item (cmdb_ci) table or its child tables. For example, if the new class is Laptops, which is a subclass of Computers, select the cmdb_ci_computer table. If the new class is a top-level class, select the cmdb_ci table.

The most commonly used parent CI types are:
- cmdb_ci - basic
- cmdb_ci_database - for databases
- cmdb_ci_app_server - for application servers
- cmdb_ci_infra_service - for infrastructure services
- cmdb_ci_endpoint_inclusion - for entry points of the inclusion type
- cmdb_ci_appl - for applications
- cmdb_ci_web_server - for web servers
- cmdb_ci_lb - for load balancers
- cmdb_ci_endpoint - for entry points

---

4. Configure how the instance determines if a discovered CI is an upgraded CI existing in the instance or a brand new CI. See [Create or edit a CI identification rule](#) for detailed procedure.

**Warning:** If there is no CI identification rule for a CI type, Service Mapping discovers CIs belonging to this type, but cannot interpret the results of the discovery process. In this case, the Now Platform rejects the discovery results for these CIs and their information is not updated.

a) On the **Identifiers** page, configure the parameters as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Use a meaningful name.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Enter the CI type.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box.</td>
</tr>
<tr>
<td>Independent</td>
<td>Clear the check box.</td>
</tr>
</tbody>
</table>

b) Configure the identifier entry parameters as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Criterion attributes</td>
<td>Select the attribute that you want to use for identifying the CI. For example, <strong>Class</strong>, <strong>Installation directory</strong>, or <strong>Serial number</strong>.</td>
</tr>
<tr>
<td>Allow null attribute</td>
<td>Select the check box.</td>
</tr>
</tbody>
</table>
5. Optional: Configure references to other CIs or non-CIs as related items for the CI type.

**Related Entry form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifier</td>
<td>The CI identifier for which this related entry belongs to. By default it is set to the identifier you previously selected.</td>
</tr>
<tr>
<td>Related table</td>
<td>A related table that references the CI that is being matched.</td>
</tr>
<tr>
<td>Referenced field</td>
<td>A referenced field in Related table that should store the referenced CI. This field always references the cmdb_ci table, or a descendent of the cmdb_ci table.</td>
</tr>
<tr>
<td>Criterion attributes</td>
<td>The set of attributes to uniquely identify the related item. Attributes can belong to the current class, or to a parent class. Click the lock icon to view, add, or remove attributes from the identification rule.</td>
</tr>
<tr>
<td>Optional condition</td>
<td>Use the Add Filter Condition and the Add “OR” Clause buttons to construct a filter to narrow the set of records that will be searched for a matching related item.</td>
</tr>
<tr>
<td>Active</td>
<td>Check box that specifies that the related entry is active.</td>
</tr>
<tr>
<td>Allow null attribute</td>
<td>If at least one criterion attribute in the related table is not null, allow to attempt matching with an identifier entry even if there are criterion attributes which are null.</td>
</tr>
<tr>
<td>Priority</td>
<td>Priority of the related entry for the specified Related table. Rules with lower priority numbers are given higher priority while matching a related item for specific related table. Related entries for the specified related table with identical priorities are applied randomly. You can keep gaps between the priority numbers, so you can assign the unused priority numbers to new entries without modifying the existing priority order.</td>
</tr>
</tbody>
</table>

6. Optional: Configure the instance to consolidate CI attributes received from different data sources correctly. See [Create or edit a CI reconciliation rule](#).

Configure the following Service Mapping-related parameters correctly:
Create entry point types for Service Mapping

An entry point is how clients access an application service. If necessary, you can create a new entry point type in addition to preconfigured entry point types in Service Mapping.

If your ServiceNow instance uses domain separation and you have access to the global domain, select the domain to which the application service belongs from the domain picker. The selected domain must be a domain without any child domains.

Role required: sm_admin or admin

Service Mapping starts the discovery and mapping process for every application service from the entry point you define for it. In addition to this, Service Mapping patterns use entry points to discover CI outbound connections.

Service Mapping includes a wide range of preconfigured entry point types that cover most commonly used applications. If your organization uses a less known or proprietary application that does not have a corresponding entry point type in Service Mapping, you must create it.

Entry points are modeled in the ServiceNow CMDB as CIs of endpoint type. Entry points are stored as records in the Endpoint (cmtb_ci_endpoint) tables.

Like any other CI type, an entry point contains several important definitions that apply to all CIs belonging to it:

- CI attributes are added as fields to the CMDB tables.

---

Note: If you do not create a CI reconciliation rule, data discovered by patterns is used to update CI attributes.

7. Configure the CI classification for your CI type:
   - For CI types representing applications, perform configuration as described in Create a Discovery process classification.
   - For CI types representing SNMP devices, perform configuration as described in Create a Discovery CI classification.

Note: There is no need to create CI classifications for hosts because these classifications are included in the base system.

8. For CI types that represent inclusions, define the hierarchy for the new CI type. Clear the Reverse Relationship Direction check box while performing this configuration. See Create dependent relationships.

9. If necessary, customize icons that represent CIs in maps. See Create or modify map icons.

---

**ServiceNow Definition form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data source</td>
<td>Select ServiceWatch.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Select the relevant CI type.</td>
</tr>
<tr>
<td>Optional condition</td>
<td>Set a condition if necessary.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Identifiers help Service Mapping and Discovery to differentiate between new and existing CIs. For example, if there is an Active Directory Forest endpoint CI type defined in the CMDB, and Service Mapping discovers an Active Directory Forest CI, it processes it using identifiers and recognizes it as an updated version of the Active Directory Forest CI that exists in the system, not a new Active Directory Forest CI. Unlike with regular CI types, identifiers for new endpoint CI types are created automatically.

CI type hierarchy. CMDB stores CI class in the form of a hierarchy where some CI classes are parents to other CI classes, who automatically derive their parent's attributes in addition to attributes you configure specifically for child CI classes. CI class hierarchy is used widely for configuring CI behavior, relationships, and display. Create standard entry points as child CIs for the endpoint CI type, which creates an extension for the cmdb_ci_endpoint table. For entry points of inclusion type create child CIs for the inclusion endpoint CI type extending the cmdb_ci_endpoint_inclusion table. In an inclusion, a server hosts applications that are treated as independent objects.

1. Navigate to Configuration > CI Class Manager.
2. To create a standard entry point, right-click Endpoint from the Class Hierarchy pane and select Add Child Class.
3. To create an entry point of the inclusion type, right-click Inclusion Endpoint from the Class Hierarchy pane and select Extend.
4. Create an entry point type using the following parameters.
   See Create a table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Entry point type name. For example, HTTP entry point.</td>
</tr>
<tr>
<td>Name</td>
<td>The table name. For example, cmdb_ci_endpoint_http.</td>
</tr>
<tr>
<td>Extends table</td>
<td>The table name of the parent CI type is automatically filled by the system:</td>
</tr>
<tr>
<td></td>
<td>• cmdb_ci_endpoint - for entry points</td>
</tr>
<tr>
<td></td>
<td>• cmdb_ci_endpoint_inclusion - for entry points of the inclusion type</td>
</tr>
</tbody>
</table>

5. Add entry point attributes on the Columns tab at the bottom of the page.
   By default the new entry point derives attributes from its parent CI, but you can modify the attributes as necessary.
6. Click Submit.

Create or modify patterns

Create or modify a discovery pattern and define its basic attributes.

Make sure that the application for which you want to create a pattern, has a corresponding configuration item (CI) type and a CI classification. If the CI type you require is not in the list, create it as described in Create CI types for Service Mapping and Discovery.
If your ServiceNow instance uses domain separation and you have access to the global domain, select the domain to which the application service belongs from the domain picker. The selected domain must be a domain without any child domains.

Role required: pd_admin

Basic knowledge of programming is desirable.

Patterns can be of infrastructure or application type. Infrastructure patterns are used only by Discovery for creating lists of devices. Application patterns serve both Service Mapping and Discovery that use the same application patterns for their purposes.

1. Navigate to Pattern Designer > Discovery Patterns.
2. Click New or select the relevant pattern from the list.
3. Define the basic pattern attributes on the Basic tab.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pattern type</strong></td>
<td>Select <strong>Application</strong> for an application pattern. It can be used both for top-down discovery performed by Service Mapping and horizontal discovery performed by Discovery. Select <strong>Infrastructure</strong> for an infrastructure pattern used for the horizontal host discovery performed by Discovery.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Enter the pattern name. This name must be unique to this pattern. Use self-explanatory names such as Apache on Unix pattern.</td>
</tr>
<tr>
<td><strong>CI type</strong></td>
<td>Select the <strong>CI type</strong> which you want this pattern to discover.</td>
</tr>
<tr>
<td><strong>Operating system</strong> (Application patterns only)</td>
<td>Select the operating system that the selected CI runs: Click <strong>All</strong> if the CI runs on more than one operating system. Or Select the relevant operating systems from the list.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Field Description

**Run Order (Application patterns only)**

For an application pattern used by Service Mapping, select the order in which this pattern always runs:
- **Before**
- **After**

Then select the other applicable pattern. This field is only relevant if a particular pattern can be confused with another pattern.

For example, both IIS and MS Exchange applications have an HTTP entry point. However, MS Exchange uses some of the components of IIS. Therefore, if the IIS pattern ran first, discovery might incorrectly identify MS Exchange as IIS. To prevent this error, in the **Run Order** field in the MS Exchange pattern definition, select **Before** and **IIS**.

**Description**

Provide a description for this pattern.

---

4. (For application patterns only) To make the MID Server run this pattern only if the process identified on a CI matches the classification criteria for this pattern, select **Enforce Process Classification**.

All simplified patterns created from generic applications have this attribute enabled. For more information about creating process classification, see [Discovery classifiers](#).

5. Define a set of identification steps for every incoming connection of a configuration item (CI).
   
a) In the **Identification Section**, click **New** and then configure the following parameters:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name for the identification section.</td>
</tr>
<tr>
<td>Entry Point Types (Application patterns only)</td>
<td>Select all relevant entry point types. Every CI has incoming connections that are referred to in Pattern Designer as entry points. You base your CI identification process on the CI entry points creating steps and defining step operations and variables for every entry point separately. For application patterns used by Discovery, enter either <strong>TCP</strong> or <strong>All</strong>.</td>
</tr>
</tbody>
</table>
## Field

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find Process Strategy (Application patterns only)</td>
<td>Select the appropriate strategy for finding the process that populates the Process variable in the Temporary Variables table.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Listening Port</strong>: The entry point is the listening port.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Target Port and IP</strong>: The entry point port communicates with another server.</td>
</tr>
<tr>
<td></td>
<td>· <strong>None</strong>: The process variable is not populated.</td>
</tr>
<tr>
<td></td>
<td>Typically, the port type of the entry point determines the strategy.</td>
</tr>
<tr>
<td>Order</td>
<td>Select a number that determines the order in which Service Mapping or Discovery use identification sections. The section with the lowest order number is used first.</td>
</tr>
</tbody>
</table>

b) Click **Save**.

c) Click the identification section name.

d) Define discovery steps as described in [Define discovery steps](#).

e) Click **Save**.

6. If necessary, create more identification sections.

7. Save the pattern.

- For application type patterns, continue with [Define the connection section](#).

### Define discovery steps

For each Identification Section and Connection Section entry that you added to the discovery pattern, define discovery steps. These steps are the basis for discovery.

Role required: pd_admin

You can choose to define these steps immediately after adding an Identification or Connection section, or you can choose to do it after adding all Identification and Connection sections.

Define an operation for every step. The type of operation dictates which parameters and variables need to be configured.

1. On the pattern form, click the relevant entry in the Identification Sections or Connectivity Sections. **Connectivity Sections** is for Service Mapping only.

   The Pattern Designer opens showing the Steps tree on the left.

   If no discovery steps have been identified for this pattern, the Untitled Step appears in the Steps tree in the left pane of the window.
2. To add a new step:
   - Click 
     to add a step above the current step.
   - Click 
     to add a step below the current step.

3. Select an operation from the list and then fill in the fields that appear for the operation.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change User</td>
<td>Use operating system credentials instead of the default administrative credentials.</td>
</tr>
<tr>
<td>Create Connection</td>
<td>Provide information about outgoing connections. This is for the connectivity section of a pattern that applies to Service Mapping only.</td>
</tr>
<tr>
<td>Filter Table</td>
<td>Filter a table according to specified criteria.</td>
</tr>
<tr>
<td>Find Matching URL</td>
<td>Find the best match for a URL in a list of URLs.</td>
</tr>
<tr>
<td>Get Registry Key</td>
<td>Query for registry keys.</td>
</tr>
<tr>
<td>Get Process</td>
<td>Search for a process according to specified criteria.</td>
</tr>
<tr>
<td>LDAP Query</td>
<td>Query an LDAP directory.</td>
</tr>
<tr>
<td>Library Reference</td>
<td>Combine a number of steps to be executed as a group.</td>
</tr>
<tr>
<td>Match</td>
<td>Match a condition and stop running the pattern if a condition is not met.</td>
</tr>
<tr>
<td>Merge Table</td>
<td>Merge two tables.</td>
</tr>
<tr>
<td>Parse Command Output</td>
<td>Extract information from the output of the command.</td>
</tr>
<tr>
<td>Parse File</td>
<td>Extract information from a file.</td>
</tr>
<tr>
<td>Parse a URL</td>
<td>Break down a URL into its components.</td>
</tr>
<tr>
<td>Operation</td>
<td>Objective</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Parse Variable</td>
<td>Extract information from a variable.</td>
</tr>
<tr>
<td>HTTP Get Call</td>
<td>Extract information from devices using HTTP protocol.</td>
</tr>
<tr>
<td>Cloud Rest Call</td>
<td>Extract information from cloud computing devices like Microsoft Azure or Amazon Web Services</td>
</tr>
<tr>
<td>Create Relation/Reference</td>
<td>Create relationships and references between CIs that were discovered within the pattern.</td>
</tr>
<tr>
<td>Put File</td>
<td>Transfer a file to a remote system.</td>
</tr>
<tr>
<td>Run an SSH script file</td>
<td>Run composite commands or sequences of commands on Unix-based hosts.</td>
</tr>
<tr>
<td>Set a parameter value</td>
<td>Set the value of a parameter.</td>
</tr>
<tr>
<td>SNMP Query</td>
<td>Execute an SNMP query.</td>
</tr>
<tr>
<td>Transform Table</td>
<td>Add computed columns to an existing table.</td>
</tr>
<tr>
<td>Unchange User</td>
<td>Switch back to the default administrative credentials.</td>
</tr>
<tr>
<td>Union Tables</td>
<td>Append two tables that share the same format.</td>
</tr>
<tr>
<td>WMI Method Invocation</td>
<td>Execute a method using WMI (Windows Management Instrumentation).</td>
</tr>
<tr>
<td>WMI Query</td>
<td>Execute a WMI query.</td>
</tr>
</tbody>
</table>

If there are custom operations, which do not come with the base system, you can select a custom operation. For more information, see [Customize pattern operations](#).

4. Specify the following discovery step settings.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precondition</td>
<td>Select this check box to add a specific criteria to the step. If the step is always performed as defined, leave this setting unchecked. For more than one condition, consider defining a step for each condition rather than multiple conditions. For more information, see <a href="#">Make a step conditional</a>.</td>
</tr>
<tr>
<td>CI Attributes</td>
<td>Table that is automatically populated with CI attribute variables that are generated when you add a CI type. This table does not support Container or Tabular variables. You can use shortcuts to enter values as described in <a href="#">Enter values and variables in patterns</a>. All variables are notated with a $ prefix and constants are formatted within double quotes. For more information, see <a href="#">Pattern variables</a>.</td>
</tr>
</tbody>
</table>
5. To add comments to any step, click the comment icon ( ), add the text in the comment text and click Post.

The comment icon changes to indicate that there is a comment associated with this pattern:

6. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

7. To delete a step from the section, select the step and click the trash can icon.

8. After you define all steps, click Save.

9. On the pattern record, click Activate to make that pattern available for use.

Click Debug to access the additional actions and to browse to and open source files rather than looking them up separately.

**Enter values and variables in patterns**

There are several ways to enter values in Pattern Designer. You can use the following types of values in patterns: strings, variables, concatenated variables, and eval() functions. Do not use GlideRecords in patterns.

**Useful shortcuts in Pattern Designer**

<table>
<thead>
<tr>
<th>To enter this value</th>
<th>Follow these steps</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter constant values, a string</td>
<td>Enclose the value in quote marks (&quot;),</td>
<td>Enter the path with quote marks (&quot; at the beginning and at the end of the string.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>To enter this value</th>
<th>Follow these steps</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter a value that can change, a variable</td>
<td>Start the variable name with the dollar sign ($).</td>
<td>Enter the variable name beginning with the dollar sign ($).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Name: $process.pid</td>
</tr>
<tr>
<td>Enter a value that can change from a variable that contains several variables</td>
<td>Enter the variable name in the following format: $&lt;container_variable&gt;.&lt;variable&gt;</td>
<td>Enter the dollar sign ($), followed by process for the container variable name, period (.), and then pid for the name of the string variable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Value: $process.pid</td>
</tr>
<tr>
<td>Enter a value from the specific field in a table</td>
<td>Enter the variable in the following format: $tabular_variable[row number].column_name</td>
<td>To use the value from the second row in the instanceID column from the IfTable variable, enter $IfTable[2].InstanceID.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>To enter this value</td>
<td>Follow these steps</td>
<td>Example</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Enter values from a specific column in a table sequentially, starting from the current row</td>
<td>Enter the variable in this format: $tabular_variable().column_name</td>
<td>To use the value from the current row in the <strong>instanceID</strong> column from the <em>IfTable</em> variable, enter $IfTable[].InstanceID.</td>
</tr>
<tr>
<td>Enter values from a specific column in a table sequentially, starting from the first row</td>
<td>Enter the variable in this format: $tabular_variable(*).column_name</td>
<td>To use the value from the first row in the <strong>instanceID</strong> column from the <em>IfTable</em> variable, enter $IfTable[*].InstanceID.</td>
</tr>
</tbody>
</table>

*Note:* When used in **Match a condition** with the **Is Not Empty** operator, the system extracts values from fields, even if some fields are empty.
<table>
<thead>
<tr>
<th>To enter this value</th>
<th>Follow these steps</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy a value or a variable into a field</td>
<td>Select the variable or the value you want to copy from the CI Attribute pane and drag it into the target field.</td>
<td>Drag the <strong>company</strong> variable from the CI Attributes pane into the Values field.</td>
</tr>
<tr>
<td>To enter this value</td>
<td>Follow these steps</td>
<td>Example</td>
</tr>
<tr>
<td>--------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Enter a variable using auto-complete</td>
<td>1. Type the dollar character ($) and the first letters of the variable name.</td>
<td>Variables have a $ prefix. Typing $P in a field displays a list of possible values beginning with &quot;P&quot;.</td>
</tr>
<tr>
<td></td>
<td>2. Select the relevant value from the list showing all currently available values that match the characters you entered.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If only one choice fits, that value is automatically entered into the field.</td>
<td></td>
</tr>
<tr>
<td>Specify complex (concatenated) values in fields</td>
<td>Enter a value, then add a plus sign (+), and then enter another value.</td>
<td>To specify the path, use the install_directory variable and extract of the actual path connected by a plus sign (+).</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> You can drag values and variables to create complex values.</td>
<td></td>
</tr>
<tr>
<td>To enter this value</td>
<td>Follow these steps</td>
<td>Example</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Use the eval() function to evaluate a string in the field</td>
<td>1. Type <code>EVAL(</code> and click the edit button next to the field.</td>
<td>To use the eval() function for JavaScript serving as a parameter value, enter the script in the EVAL Script Editor window.</td>
</tr>
<tr>
<td></td>
<td>2. In the EVAL Script Editor window, ensure that the JavaScript mode is on or click Other for other types of scripts like Groovy.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Type the script in the editor pane.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4. If necessary, use the Search, Replace, and Format buttons to modify the script.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. Click <strong>Apply</strong>.</td>
<td></td>
</tr>
</tbody>
</table>

**Activate pattern Debug mode**

Working in Debug mode, Pattern Designer performs all operations as you configure them. It allows you to see results immediately.

Before starting this procedure, verify that the Debug mode is not activated. If the Debug mode is activated, the debug button appears with a green dot:

**Role required:** pd_admin

**Basic knowledge of programming is desirable.**

Many operations require that you enter a specific value from a specific source, for example, a particular value, or a particular location and delimiter, in a particular file. If you are not in Debug mode, you must type details such as path and file name, or an actual relevant value or related information. However, if you work in Debug mode, you can browse to and open information sources such as files, and then select specific values from those files.
Service Mapping and Discovery share a set of preconfigured patterns that cover most of the commonly used devices and applications. Patterns can be of infrastructure or application type. Infrastructure patterns are used only by Discovery for creating lists of devices. Application patterns serve both Service Mapping and Discovery that use the same application patterns for their purposes.

Service Mapping and Discovery share patterns, but execute them differently. Discovery runs infrastructure, application, and cloud resource patterns for horizontal discovery, while Service Mapping runs application patterns for the top-down discovery. When you activate Debug mode, you choose how pattern steps are run: as if by Discovery or as if by Service Mapping.

When you activate the Debug mode, Pattern Designer performs an operation in a pattern step and displays discovered values in the Temporary Variables pane.

When you activate the Debug mode, the following additional actions become available:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Temporary Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parse URL</td>
<td>install_folder</td>
</tr>
</tbody>
</table>
| Source: $entry_point.url | C:\Windows\sys...
| Target: $url_table     | name: IIS 7.5
|                 | version: 7.5         |
## Debug Mode Actions

<table>
<thead>
<tr>
<th>Action</th>
<th>Allows you to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search assistant</td>
<td>Search within files or the registry.</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>Perform additional configuration to enable this feature as described in</td>
</tr>
<tr>
<td></td>
<td>Configure Search Assistant for Windows.</td>
</tr>
</tbody>
</table>

| Command Prompt  | Check details of file structure, setup, or parameters and values inside specific files on a host before using this information in step operations. |

| Test            | Test the current step and populates variables with discovered values.         |

Activate the Debug mode for each browser window or tab separately. For example, if you have the same pattern section open in two or more tabs, you must activate the Debug mode in each tab.
Discovery and Service Mapping use algorithms to select a MID Server for discovery. While working in Debug mode, you can either rely on the MID Server chosen by the algorithm or select a different MID Server from the list. This list contains only MID Servers that are valid, running, and having the matching IP address range for the pattern step. Discovery and Service Mapping use the same MID Server for the entire Debug session, whether the algorithm selected this MID Server or you chose it.

1. In the Pattern Designer, click **Debug Mode**.
   The Debug Identification Section window is displayed.

2. Fill in the required details for the MID Server entry point:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select MID Server</td>
<td>(Optional) Select the MID Server for this Debug session.</td>
</tr>
<tr>
<td>Debug Type</td>
<td>Select <strong>Top down</strong> for performing top-down discovery with Service Mapping or <strong>Horizontal</strong> for performing horizontal discovery using Discovery.</td>
</tr>
<tr>
<td>Select Entry Point Type</td>
<td>Select the entry point type for the discovered CI. Entry point parameters depend on the type you select.</td>
</tr>
<tr>
<td>URL</td>
<td>Enter the URL of the CI you are discovering.</td>
</tr>
<tr>
<td>Host Name</td>
<td>Enter the host name of the server hosting the discovered application.</td>
</tr>
<tr>
<td>Comments</td>
<td>Any considerations or notes.</td>
</tr>
<tr>
<td>Use original IP</td>
<td>Select the check box and enter the IP for this entry point.</td>
</tr>
</tbody>
</table>

3. The Debug Identification Section window, click **Connect**.
   The Debug mode is activated. The **Debug** button shows the green dot. For connection sections the Debug Mode button also shows the Debug type you chose:

   ![Debug Mode - Top down](image)

   When you select a different section in the same pattern, the Debug mode stays activated if it can use the entry point you defined for the previous step.

4. Optional: If you click another section, to which Pattern Designer cannot connect using the entry point for the previous step, define the entry point again as described in step 2.

5. Optional: To activate the Debug mode for the same pattern section to which you navigated from the pattern form in a separate browser window or tab, define the entry point again as described in step 2.

6. To deactivate the Debug mode, click the **Debug Mode** button.

**Make a step conditional**
If necessary, create a condition that defines how Discovery and Service Mapping execute a pattern step.

Basic knowledge of programming is desirable.

Create a pattern or select a pattern that you want to modify.
Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

Role required: pd_admin

You can define a single or multiple prerequisite conditions. For example, you can define that a step is run only during the horizontal or top-down discovery using the pattern_runtime_mode variable.

1. Click the Precondition check box next to the Operation list.
   Not all operations allow you to use preconditions.
2. In the first condition field, enter the required value.
   For example, enter the actual string or a variable name.

3. Select an operator from the list.
   If you select Is Empty, the second field is rendered irrelevant and disappears.
4. To add more conditions, click the plus icon and define the criteria.
5. If you create multiple conditions, define if this pattern must meet all or any of them: From the Meet list, select All or Any.
6. Define if the criteria must be satisfied or not for the step operations to run: From the If Condition is list, select True or False.

You create a pattern step to filter data in the cluster string table to receive a table containing only cluster names. You can set a precondition to filter the table only if the cluster string variable is not empty.
Customize pattern operations
As part of creating or modifying a discovery pattern, modify existing custom operations that come as part of the base system or add new ones.

Practical knowledge of Java scripting is required.

Role required: pd_admin

For standard pattern operations, you can define only input parameters. However, there are custom operations for which you can define the business logic and the mechanism of the operation itself in addition to input parameters. For example, you can make a parameter mandatory or define which input parameters to use.

The following custom operations are available:

Cloud REST Call
Extracts information from configuration items of the PaaS (Platform-as-a-Service) type, such as Microsoft Azure or Amazon Web Services. This Java-based custom operation is part of the base system.

HTTP Get Call
Extracts information from configuration items (CIs), which use the HTTP protocol. This custom operation is part of the base system.

Cloud REST Query
Extracts information from configuration items of the PaaS (Platform-as-a-Service) type, such as Microsoft Azure or Amazon Web Services. This Java-script based custom operation is available only after downloading patterns version 1.0.24 or later from ServiceNow Store. Use this operation instead of the Cloud REST Call operation.

In addition to these custom operations, you can create your own operations to serve the needs of your discovery process. Custom operations created by you appear in the list of operations along with operations, which are part of the base system.

1. Navigate to Pattern Designer > Custom Operations.
2. To add a custom operation, click **New**.

Or

3. To modify an existing custom operation, click its name in the table.

4. For the new custom operation, enter a name describing the new operation in the **Name** field. For example, if the purpose of this operation is to extract information using a certain protocol, you can name it NetApp protocol query.

5. Enter or modify the operation purpose or description in the **Description** field.

6. Write the Java script in the **Script** pane to define the business logic of the operation. The script must comply to the following guidelines:

   - Create variables for operation parameters using the dollar sign in front of the variable name, for example, `$fileName`.
   - `CTX` is an object containing all the information which resulted from a pattern execution.
   - Use `rtn` to indicate the string, which is the result of your custom operation.

   For example:

   ```java
   ms.info('message from Custom operation MIDLogWrite ' + $logMessage);
   ```

   Where `ms.info` is the MID Server log. The `$logMessage` variable refers to the message that is created inside this MID Server log.

7. In the **Custom Operation Parameters** related list, define the input parameters you used in the Java script.

   For example, if you used a variable for the log message file in the Java script, you must also define the parameter for this variable:
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Enter a short and descriptive label for the parameter. The label shows in the table. For example, netapp_query.</td>
</tr>
<tr>
<td>Name</td>
<td>Enter a short and clear name for the parameter without spaces, for example, NetAppquery. This name appears in the list of operations on the pattern step page.</td>
</tr>
<tr>
<td>Default value</td>
<td>If relevant, enter the default value. For example, POST.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Set to true if this parameter is mandatory for this operation. Otherwise, set to false.</td>
</tr>
<tr>
<td>Order</td>
<td>Define the order in which parameters appear on the operation page. Use natural numbers, where 1 means this parameter appears at the top of the operation page.</td>
</tr>
</tbody>
</table>
## Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain</td>
<td>For domain-separated environments, specify the name of the domain for which this parameter is relevant.</td>
</tr>
<tr>
<td>Validation Regex</td>
<td>Specify a Java regular expression to validate the parameter with. During discovery process, the MID Server validates the operation result. The Pattern Designer module also runs validation when you click <strong>Test</strong> on the pattern step page.</td>
</tr>
</tbody>
</table>

**Explore remote host**

Check details of file structure, setup, or parameters and values inside specific files on a host before using this information in step operations.

Basic knowledge of programming is desirable.

1. Navigate to **Pattern Designer > Discovery Patterns**.
2. Select a pattern from the Discovery Patterns list.
3. Select an identification or connection section.

4. Switch to the debug mode as described in **Activate pattern Debug mode**.

Role required: pd_admin
Part of creating or modifying patterns is defining step operations that use information about the host on which they run. You can verify that the information you are going to use in operations is correct, by exploring hosts.

You can explore both remote hosts and MID Servers.

1. Click **Command Prompt** next to the Debug Mode button.
   The Command Prompt window opens.

2. Enter the command.
   To make the platform apply applicative credentials while executing this command, use placeholders for credentials using the following syntax:
   - $$username$$ - for the user name
   - $$password$$ - for the password

   For example, a parsing command for a Microsoft SQL Server uses credential placeholders:
   ```
   "su - " + $userid + " -c " + $ExecutableDir + "mysql --user=" + "$username$$" + " --password=" + "$password$$"
   ```

3. Optional: Click **Advanced Details**.

4. Optional: Specify the command and parameters:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Execute Mode</strong></td>
<td>Select the relevant option of running this command:</td>
</tr>
<tr>
<td></td>
<td>• Default (Remote) - on a remote device</td>
</tr>
<tr>
<td></td>
<td>• Local Script - on the MID Server</td>
</tr>
<tr>
<td></td>
<td>• Windows Service - on a remote Windows server</td>
</tr>
<tr>
<td><strong>CI Type</strong></td>
<td>If necessary, select the CI type whose credentials the platform applies while running the command.</td>
</tr>
<tr>
<td><strong>Host</strong></td>
<td>Enter the management IP address of the host on which to run the command. You can also leave this value empty.</td>
</tr>
</tbody>
</table>

5. Click **Run Command**.
   The result is displayed in the Output pane.

6. Click **Close** when finished checking the result.

**Pattern variables**

You use variables in discovery patterns to refer to parameters or attributes of the CI that the pattern discovers.

There are several kinds of variables used in discovery: global variables, CI attribute variables, and temporary variables.
### Characteristics

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Global variable for which you create a pattern. <strong>entry point</strong></th>
<th>CI attributes</th>
<th>Temporary variables</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Contains information about the connection which serves as an entry point for the CI for which you create a pattern. <strong>process</strong></td>
<td>A variable can be: • Scalar: A single string • Tabular: A table, where each cell is a scalar variable. • Vector: A single, unnamed column with as many rows as needed</td>
<td>Scalar, tabular, or vector • Scalar: A single string • Tabular: A table, where each cell is a scalar variable. • Vector: A single, unnamed column with as many rows as needed</td>
</tr>
</tbody>
</table>

#### Type/structure

| Type/structure | Container variable: a variable can hold any combination of single strings and tabular variables. | A variable can be: • Scalar: A single string • Tabular: A table, where each cell is a scalar variable. • Vector: A single, unnamed column with as many rows as needed | Scalar, tabular, or vector • Scalar: A single string • Tabular: A table, where each cell is a scalar variable. • Vector: A single, unnamed column with as many rows as needed |

#### Origin

| Origin | Preconfigured in Service Mapping. | Service Mapping derives them from a CI type definition. | You create these variables while defining operations for pattern steps. |

| Modifiable in Pattern Designer | No | No | Yes |

Some process variables retrieve values from the CMDB. Run the horizontal discovery prior to performing the top-down discovery to use these variables.

**Note:** The **Pre CMDB process** variable is present for lightweight identification sections or for identification sections of patterns for discovery of load balancers.

Pattern Designer displays different kinds of variables in different areas of its interface.
Always prefix variables with the dollar symbol ($) which indicates variables, but is not actually a part of the variable name. For example, if you specify $Abc as the variable name, the actual name of the variable is Abc.

Change credentials to non-default
As part of creating or modifying a discovery pattern, you can define a **Change user** operation to make Service Mapping or Discovery use SSH or Windows credentials instead of the default administrative level credentials. This operation is relevant only for configuration items (CIs) hosted on Windows or Unix operating systems.

- Ensure that the OS credentials you want to use instead of the default credentials are correctly configured. Configure the credential alias to the OS CI type whose credentials you want to use. For operational information on how to create credentials, see [SSH credentials](#) and [Windows credentials](#).
- Navigate to the relevant pattern step:
  1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.
2.

Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
You can change the default credentials to any appropriate credentials belonging to a different CI type on the same or a different host.

1. Select Change user from the Operation list in one of the following locations.
2. Select the Use Different CI Type check box.
3. From the list, select the CI type whose credentials you want to use.
4. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Business Integration Software</td>
</tr>
<tr>
<td>CI Type</td>
<td>IBM WebSphere MQ Queue</td>
</tr>
<tr>
<td>Pattern</td>
<td>WMQ Queue Unix Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Local queue connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>3. Change user credentials</td>
</tr>
</tbody>
</table>

As part of discovery of an IBM WebSphere MQ Queue, you change credentials to IBM WebSphere MQ. It allows you to manipulate local IPs to discover connections on the local queue.

- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.

Provide connection information
As part of creating or modifying a discovery pattern, use the Create connection operation to provide information about an outgoing connection. This operation is only available for connection sections.

Navigate to the relevant step:

1. On the pattern form, select the relevant connection section.
2. Select the relevant step or create a pattern step by clicking .

Basic knowledge of programming is desirable.

Role required: pd_admin

Always start with this step when creating a connection section entry for a new pattern.

1. Select **Create Connection** from the **Operation** list.

2. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Select Connection Type       | - **Application Flow**: Used between two applications (can be of the same type).
                                  - **Cluster**: Used for connections to CIs of the cluster type. Specify a cluster name.
                                  - **Inclusion**: Used for connections to an object that is included in the current object. For example, a connection from J2EE to EAR, and a connection from IIS to a website.
                                  - **Storage Flow**: Used for connections between configuration items (CIs) of host type and devices of storage type. |
| Select Entry Point           | Select the entry point type from the list. For more information, see: **Entry point attributes**. |
| Enter Connection Attributes  | Define attributes by either entering actual values or variables.             |
| Select Target CI Type        | Select the target CI type from the list.                                     |
| Is hidden                    | Select the check box if the connection should be hidden (that is, not shown in the user interface), but is used for continuing the discovery flow. |
| Is traffic based             | Select the check box to use the traffic-based discovery method for this connection. |

3. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Business Integration Software</td>
</tr>
<tr>
<td>CI Type</td>
<td>IBM WebSphere MQ Queue</td>
</tr>
<tr>
<td>Pattern</td>
<td>WMQ Queue Unix Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Alias queues connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>6. Create outgoing connection to alias queues</td>
</tr>
</tbody>
</table>

To create a connection from Microsoft Exchange CAS to Exchange Mailboxes, define the **Create Connection** operation as follows:
• Continue editing the pattern by adding a new step and defining its operation or
• Finalize the pattern.

**Merge tables**
As part of creating or modifying a discovery pattern, you can use the **Merge Table** operation to merge content from two source tables into a target table.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click ![add step icon](image) to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

Use this operation to unify information from different sources.
For example, during discovery of connections of IBM WebSphere Portal to IBM WebSphere MQ, you can merge the table containing queue names with the table containing JNDI reference names.

1. Select **Merge Table** from the **Operation** list.
2. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>First table</td>
<td>Specify the name of the first source table.</td>
</tr>
<tr>
<td>Second table</td>
<td>Specify the name of the second source table.</td>
</tr>
<tr>
<td>Target table</td>
<td>To create a table to contain merged data, define the name of the new table.</td>
</tr>
<tr>
<td>Unmatched values</td>
<td>For tables that meet the merge criteria, select an action from the list for unmatched rows:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Keep</strong>: If merge criteria is met for any row, it merges all rows from both source tables into the target table.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Remove</strong>: Merges only matching rows from both source tables into the target table, and excludes non-matching rows.</td>
</tr>
</tbody>
</table>

3. To merge tables based on one matching field value:
   a) Click **Field matching**.
   b) In the **First Table Field**, enter the value from the first source table.
   c) In the **Second Table Field**, enter the value from the second source table.

4. To merge tables based on multiple matching field values:
   a) Click **Condition**.
   b) In the first condition field, enter the required value.
      For example, enter the actual string or a variable name.
      You can use variables including values from tabular variables as described in *Enter values and variables in patterns*. 
c) Select an operator from the list.

d) To add more conditions, click the plus icon and define the criteria.

e) If you create multiple conditions, define if this pattern must meet all or any of them: From the *Meet* list, select *All* or *Any*.

5. If in Debug mode, test the step by clicking *Test* and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Application Servers</td>
</tr>
<tr>
<td>CI Type</td>
<td>Websphere Portal (cmdb_ci_appl_websphere_portal)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Websphere Portal On Linux</td>
</tr>
<tr>
<td>Section</td>
<td>EAR TO MQ Connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>8. Merge the q factories with the jndi ref names</td>
</tr>
</tbody>
</table>

To merge tables containing queue names with JNDI reference names, use the *Merge Table* operation as follows:
• Continue editing the pattern by adding a new step and defining its operation or finalize the pattern.

Find a matching URL
As part of creating or modifying a discovery pattern, you can use the Find Matching URL operation to find the best match for the main URL and create a variable to hold it.

• Make sure that you know the main URL of the application.
• Navigate to the relevant pattern step:
  1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.
  2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
Some application CIs have configuration files containing lists of possible URLs. You can search such lists to find the match for the URL that allows connecting to a configuration item (CI).

1. Select Find matching URL from the Operation list.
2. Fill in the fields, as appropriate.
### Field Description

<table>
<thead>
<tr>
<th>Values</th>
<th>Specify the table and records containing potential URLs. You can use values from variables, including temporary tabular variables: from a specific field or a specific column in a table sequentially, starting from the first row. For more information, see Enter values and variables in patterns.</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td>Specify the URL for which you are seeking the best match.</td>
</tr>
<tr>
<td>Target Variable</td>
<td>Specify the variable name to contain the best match.</td>
</tr>
</tbody>
</table>

| 3. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected. |

### This operation is used in This item

<table>
<thead>
<tr>
<th>Hierarchy</th>
<th>Applications &gt; Application Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Type</td>
<td>Jboss (cmdb_ci_app_server_jboss)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Jboss pattern</td>
</tr>
<tr>
<td>Section</td>
<td>http identification section</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>43. best match</td>
</tr>
</tbody>
</table>

To filter the table on Jboss and to create a variable to hold the best match for the Jboss entry point URL, define the **Find Matching URL** operation as follows:

- Continue editing the pattern by **adding a new step and defining its operation** or
- **Finalize the pattern**.

**Filter a table**

As part of creating or modifying a discovery pattern, you can use the **Filter table** operation to search a source table for a specified value. If found, values are logged in a specified target table.
Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

1. Select **Filter Table** from the **Operation** list.
2. Fill in the fields, as appropriate:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source table</td>
<td>Name of source table to be searched. Use drag-and-drop or auto-complete to copy table names, fields, and values from the Temporary Variables and CI Attributes tables.</td>
</tr>
<tr>
<td>Target table</td>
<td>Name of target table to contain the values that were found.</td>
</tr>
</tbody>
</table>

3. Define the condition for this operation:
   a) In the first condition field, enter the required value. For example, the table name and the column from which to retrieve the values.

You can also use variables including values from tabular variables as described in [Enter values and variables in patterns](#).
b) Select an operator from the list.

c) Enter the string that Pattern Designer uses to filter values.

d) To add more conditions, click the plus icon and define the criteria.

e) If you create multiple conditions, define if this pattern must meet all or any of them: From the Meet list, select All or Any.

4. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Application Servers</td>
</tr>
<tr>
<td>CI Type</td>
<td>WebSphere Portal (cmdb_ci_appl_webspere_portal)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Websphere Portal On Linux</td>
</tr>
<tr>
<td>Section</td>
<td>DB2 JDBC connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>8. Prepare different lists of connections per db type</td>
</tr>
</tbody>
</table>

To identify IBM WebSphere connections to Oracle applications, use the Filter Table operation as follows:

- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.
Get a process
As part of creating or modifying a discovery pattern, use the **Get process** operation to search for a specific process to store in a tabular variable.

- Navigate to the relevant pattern step:
  1. On the pattern form, select the relevant identification section for Discovery.
     Alternatively, select the relevant identification or connection section for Service Mapping.
  2. Select the relevant pattern step or click \( \text{Add Step} \) to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

You can manually specify the filtering criteria, or you can select a process from the list of all processes on the system. The values of the selected process are used to populate the filtering fields. Modify these criteria as needed (for example, to delete irrelevant criteria).

Processes that satisfy the specified filtering criteria are placed in a tabular variable whose name you specify. This tabular variable appears in the **Temporary Variables** table.

1. Select **Get process** from the **Operation** list.
2. If in the debug mode, perform the following steps:
   a) Click **Browse Process** to open a form containing a list of processes.
   b) Select a process and click **OK**. Filtering criteria are populated with values from the selected process.
3. If not working in the debug mode, define the field values as needed:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process ID</td>
<td>Enter the process ID. Not recommended since this parameter can be modified.</td>
</tr>
<tr>
<td>Command Line</td>
<td>Enter the command line or a string which is part of it. For example, you can use &quot;bw&quot; to find &quot;bwengine.&quot;</td>
</tr>
<tr>
<td>Working Directory</td>
<td>Enter the working directory for the process. Not recommended, since this parameter can vary on different hosts.</td>
</tr>
<tr>
<td>Parent Process</td>
<td>Enter the process which is the parent of the process that you want to extract. Not recommended since this parameter can be modified or can extract many irrelevant sub processes.</td>
</tr>
<tr>
<td>Port</td>
<td>Enter the port on which the process runs. Not all processes are based on ports.</td>
</tr>
</tbody>
</table>

4. In the **Specify Target Variables** field, specify the name for the tabular variable to hold the list of processes that satisfy the filtering criteria.

You can also enter a value from the specific field in a tabular variable as described in **Enter values and variables in patterns**.

5. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
This operation is used in | This item
---|---
Hierarchy | Application> Infrastructure Service > Web Server
CI Type | Microsoft Iis Web Server (cmdb_ci_microsoft_iis_web_server)
Pattern | IIS
Section | Identification for HTTP(S) entry point type(s) for IIS6 second logic
Step number and Name | 40. Get IIS process

Use the Get Process operation to get all Internet Information Services worker processes (w3wp.exe) running on the Windows Server and keep the results in the table variable iis_process.

- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.

Get a registry key
As part of creating or modifying a discovery pattern, you can use the Get registry key operation to retrieve and select registry key attributes to store in a table.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click \[+\] to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
This operation is relevant only for Windows.

1. Select \textbf{Get registry key} from the \textbf{Operation} list.
2. If working in the Debut Mode, perform the following steps:
   a) Click \textbf{Browse} and select the registry key.
      The selected key path is placed in the \textbf{Registry key path} field. A form opens and displays a list of keys next to the tree.
      You can use variables. You can also enter a value from the specific field in a tabular variable as described in \textit{Enter values and variables in patterns}.
   b) Select the key to show the attributes and click \textbf{OK}.

3. If not working in the Debug Mode, specify the registry key path in the \textbf{Registry key path} field.
4. Select the relevant option from the \textbf{Build Variables-Keys Table} list:
   \begin{itemize}
   \item Select \textbf{By Using All Keys From The Registry Directory}, and enter the name of the table to contain the keys and variables.
   \item Select \textbf{By Building The Table From The Browser or Manually}, and specify the keys on which you want to build the table. If there is more than one key, it creates a table to hold variables.
   \end{itemize}
5. Define the name of the table to which you want to save the operation result.
6. Select \textbf{Terminate} to stop discovery if no results are found.
7. If in Debug mode, test the step by clicking \textbf{Test} and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Application &gt; Infrastructure Server &gt; Web Server</td>
</tr>
<tr>
<td>CI Type</td>
<td>Microsoft iis Web Server (cmdb_ci_microsoft_iis_web_server)</td>
</tr>
<tr>
<td>Pattern</td>
<td>IIS</td>
</tr>
<tr>
<td>Section</td>
<td>Identification for HTTP(s) entry point(s) for IIS6</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>2. get version from registry</td>
</tr>
</tbody>
</table>

Get registry keys and values of "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft \inet\St" and save the value in the 'version' variable.
Continue editing the pattern by adding a new step and defining its operation or finalize the pattern.

Define an LDAP query
As part of creating or modifying a discovery pattern, you can use the LDAP query operation to query an LDAP directory.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

1. Select LDAP query from the Operation list.
2. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base DN</td>
<td>Specify the point from where the LDAP server searches for users.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Specify the name of the server that hosts the LDAP server.</td>
</tr>
<tr>
<td>Port</td>
<td>Specify the port name.</td>
</tr>
<tr>
<td>Query</td>
<td>Specify the query parameters.</td>
</tr>
</tbody>
</table>
| Scope | Specify the query scope from the list:  
  - **Sub Tree**: the object and all descendants.  
  - **Object**: the object only.  
  - **One Level**: the object and one level below. |

3. In the **Variable Table**, click **Add** and define the table name and column name to hold the query result.  
   For multiple results, click **Add** for each additional column that is needed.  
4. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Application &gt; Exchange Server Component</td>
</tr>
<tr>
<td>CI Type</td>
<td>ExchangeBackEndServer</td>
</tr>
<tr>
<td>Pattern</td>
<td>ExchangeBackEndServer On Windows Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Storage connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>2. Query ldap - msExchStorageGroup.msExchESEParamSystemPath</td>
</tr>
</tbody>
</table>

To find storage devices connected to the Exchange BackEnd Server, you can use the following LDAP query: 
```
“(objectclass=msExchStorageGroup)“
```

The operation result is stored in a variable table with the following fields:  
msExchESEParamSystemPath and msExchStorageGroupFiles1v. You can use variables to specify Base DN that indicates where to begin the search, as well as the host name and port.
As part of creating or modifying a discovery pattern, you can use the **Match** operation to specify conditions that the discovery process must meet to continue. If these conditions are not met, the discovery process stops.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click ![Add Step](image) to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

While the discovery process always stops if the specified conditions are not met, you can control if the system regards this result as discovery error or not. In some cases you may run a pattern containing the match operation and expect the result not to match conditions. For example, in cloud environments different data centers may have different sets of devices and applications. You run a pattern for a certain application knowing that you do not find it on all data centers.
If you define that the lack of match is not a discovery error, the discovery log shows the message instead of the discovery error. Customize the message to provide useful information about why this result is expected.

1. Select **Match** from the **Operation** list.
2. In the first condition field, enter the required value.
   You can use an actual string or a variable. You can also use values from temporary tabular variables: from a specific field or a specific column in a table sequentially, starting from the first row. For more information, see **Enter values and variables in patterns**.

3. Select an operator from the list.
4. If necessary, enter the required value in the second condition field.
5. To add more conditions, click the plus icon and define the criteria.
6. If you create multiple conditions, define if this pattern must meet all or any of them: From the **Meet** list, select **All** or **Any**.
7. To consider the lack of match as not a discovery error:
   a) Select **Expected** from the **Termination type** field.
   b) Enter the text for the discovery message to provide useful information about the expected result.
      You can use variables in the discovery message. For example, you can enter the following text:
      
      The discovery was stopped due to the following error: $error.

8. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Business Integration Software</td>
</tr>
<tr>
<td>CI Type</td>
<td>IBM WMB HTTP Listener</td>
</tr>
<tr>
<td>Pattern</td>
<td>WMB HTTP Listener On Unix Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Identification for HTTP</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>1. Check process name to match http lstnr</td>
</tr>
</tbody>
</table>
During discovery of an IBM WMB HTTP Listener, use the Match operation to check the process name.

![Match operation screenshot]

- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.

Parse command output
As part of creating or modifying a discovery pattern, you can use the Parse command output operation to extract information from the command output and to save the operation result in a variable. You can also save the whole of command output as a variable.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click ![Add step icon] to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
1. Select Parse command output from the Operation list.
2. Specify the command in Set Command Details.
   You can use variables. You can also enter a value from the specific field in a tabular variable as described in Enter values and variables in patterns.
   You can concatenate multiple commands.

Important: Avoid entering a specific path to a location or file because it can be different on different operating systems. You should use variables for paths.

To make the Now Platform apply applicative credentials while executing this command, use placeholders for credentials using the following syntax:
- $$username$$ - for the user name
- $$password$$ - for the password

For example, a parsing command for a Microsoft SQL Server uses credential placeholders:
"su - " + $userid + " -c '" + $ExecutableDir + "mysql --user=" + "$username$$" + " --password=" + "$password$$"

3. To change the execution mode or credentials, click Advanced and fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execute Mode</td>
<td>Select the relevant option for running this command:</td>
</tr>
<tr>
<td></td>
<td>· Default (Remote) - on the remote device.</td>
</tr>
<tr>
<td></td>
<td>· Local Script - on the MID Server</td>
</tr>
<tr>
<td></td>
<td>· Windows Service - on the service running on the remote Windows server</td>
</tr>
<tr>
<td>CI Type</td>
<td>If necessary, select the CI type whose credentials the platform applies while running the command.</td>
</tr>
</tbody>
</table>

4. To save the whole of the command output as a variable, select NONE from the Define Parsing list and define the variable as described in 7.

5. Select the parsing strategy from the Define Parsing list.

<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>Horizontal file parsing strategy (not vertical). You can use this parsing strategy only for text files. For more information, see Parse text from a horizontal file.</td>
</tr>
<tr>
<td>LDAP file</td>
<td>Retrieve text from a structured text file where each set of data spans multiple lines. For more information, see Parse text from a vertical file.</td>
</tr>
<tr>
<td>XML file</td>
<td></td>
</tr>
<tr>
<td>INI file</td>
<td></td>
</tr>
<tr>
<td>Properties file</td>
<td></td>
</tr>
<tr>
<td>JSON file (custom)</td>
<td></td>
</tr>
<tr>
<td>Vertical File</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>After Keyword</td>
<td>Retrieve text directly following a specific keyword. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Command Line Java Style</td>
<td>Retrieve the value of a command-line parameter using Java-style parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Command Line Unix Style</td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Position From End</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
</tbody>
</table>
### Parsing strategy

<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position From Start</td>
<td>Retrieve text specified by its position from the beginning of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Retrieve text specified by a regular expression. This option requires familiarity with Regex Java syntax. For more information, see Parse text using a regular expression.</td>
</tr>
<tr>
<td>Delimited Text</td>
<td>Retrieve text specified by delimiters and position within the line (the most common way to retrieve text from generic text files). See Parse text using delimited text.</td>
</tr>
</tbody>
</table>

6. If working in Debug mode, perform the following steps:
   a) Click **Run Command** to see the result in the **Output** pane.
   b) In the **Output** pane, mark text or symbols that you want a variable to contain.
   c) In the variable name box, enter the name for the new variable, for example ‘process_name’.
   d) Press **Enter**.

   The new variable is added in the **Variables** pane.

7. If you are not working in Debug Mode, define the parsing criteria as follows:
   a) In the Variables pane, click **Add Variable** and enter the name for the new variable.
   b) Click the **Advanced** icon.
c) Click **Add Column Label**.

d) Enter the parsing query.

e) Enter a value for the delimiter.

f) Enter a value for the position.

8. Select **Use Cache** to save the operation results in cache on the MID Server. Use cache to optimize discovery and avoid creating unnecessary load on central shared components, such as load balancers. The base system keeps operation results in cache for an hour.

9. Select **Terminate** to stop discovery if no results are found.

10. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

In this example, the Tibco parser script contained in the `$TibcoParser` variable runs on the target host. The parsing method is delimited text with the `=` delimiter. The result of the parsing is the path to the directory containing all configuration files for Tibco Business Works and Tibco Business Works Process. This step creates a temporary variable to hold the result of the parsing.
Continue editing the pattern by adding a new step and defining its operation or Finalize the pattern.

Parse a file
As part of creating or modifying a discovery pattern, you can use the Parse file operation to extract information from a file and create variables to contain the extracted information.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
Alternatively, you can use traffic-based connections to create a parse file step in the relevant CI pattern as described in *Fine-tune patterns using traffic-based discovery*.

1. Select **Parse file** from the **Operation** list.
2. Specify the file path in **Select file**.
   
   You can use variables. You can also use values from a temporary tabular variable: from a specific field or a specific column in a table sequentially, starting from the first row. For more information, see *Enter values and variables in patterns*.

   To specify the actual file, click **Browse**, navigate to the file, and click **Select**.

   **Important:** Avoid entering a specific path to a location or file because it can be different on different operating systems. You should use variables for paths.

3. Select the relevant parsing strategy from the **Define Parsing** list and define parsing criteria.

<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>Horizontal file parsing strategy (not vertical). You can use this parsing strategy only for text files. For more information, see Parse text from a horizontal file.</td>
</tr>
<tr>
<td>LDAP file</td>
<td>Retrieve text from a structured text file where each set of data spans multiple lines. For more information, see Parse text from a vertical file.</td>
</tr>
<tr>
<td>XML file</td>
<td>Retrieve the value of a command-line parameter using Java-style parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>INI file</td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Properties file</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>JSON file (custom)</td>
<td>Retrieve text specified by its position from the beginning of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Retrieve text specified by a regular expression. This option requires familiarity with Regex Java syntax. For more information, see Parse text using a regular expression.</td>
</tr>
</tbody>
</table>
Delimited Text

Retrieve text specified by delimiters and position within the line (the most common way to retrieve text from generic text files). See Parse text using delimited text.

4. Select Terminate to stop discovery if no results are found.
5. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

You can use the Parse File operation to extract data on prices and authors from an xml file using the XML File parsing strategy.

- Continue editing the pattern by adding a new step and defining its operation or Finalize the pattern.

Parse a URL
As part of creating or modifying a discovery pattern, you can use the Parse URL operation to break down a URL to the component level.
Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin

The URL parsing breaks a URL into the following components: protocol, hostname or IP, port, path and file.

1. On the Identification or Connectivity Sections form, select Parse URL from the Operation list.
2. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Specify the URL:</td>
</tr>
<tr>
<td></td>
<td>• Enter a value that can change, a variable. Type the dollar character ($) and the first letters of the variable name. For example, $ldap_url.</td>
</tr>
<tr>
<td></td>
<td>• Specify complex (concatenated) values in fields. Enter a value, then add a plus sign (+), and then enter another value. For example, $install_directory+&quot;conf/httpd.conf&quot;.</td>
</tr>
<tr>
<td></td>
<td>• Enter constant values, a string. For example, &quot;/opt/ibm/mqsi/7.0/bin/&quot;.</td>
</tr>
<tr>
<td></td>
<td>• Enter a value from a tabular variable: from a specific field or a specific column in a table sequentially, starting from the first row. For more information, see Enter values and variables in patterns.</td>
</tr>
</tbody>
</table>

Important: Avoid entering a specific path to a location or file because it can be different on different operating systems. You should use variables for paths.

| Target     | Specify the table to hold the results. Use variables, complex values or strings as described above. |

3. Select Terminate to stop discovery if no results are found.
4. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Application Servers</td>
</tr>
<tr>
<td>CI Type</td>
<td>Apache Web Server</td>
</tr>
<tr>
<td></td>
<td>(cmdb_ci_apache_web_server)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Apache On Unix Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Create LDAP connections</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>3. parse ldap url</td>
</tr>
</tbody>
</table>
To break an LDAP directory URL to components and save the results in a new table, use the Parse URL operation as follows:

- Continue editing the pattern by adding a new step and defining its operation or Finalize the pattern.

### Parse a variable

As part of creating or modifying a discovery pattern, you can use the Parse variable operation to extract information from a variable and to store it in a variable table.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

1. Select Parse variable from the Operation list.
2. Define the variable that you wish to parse in the Enter Variable field.
   You can use regular variables. You can also use values from a temporary tabular variable: from a specific field or a specific column in a table sequentially, starting from the first row. For more information, see Enter values and variables in patterns.
3. Select the relevant parsing strategy from the Define Parsing list and define the parsing criteria.
<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>Horizontal file parsing strategy (not vertical). You can use this parsing</td>
</tr>
<tr>
<td>LDAP file</td>
<td>strategy only for text files. For more information, see Parse text from a</td>
</tr>
<tr>
<td>XML file</td>
<td>horizontal file.</td>
</tr>
<tr>
<td>INI file</td>
<td></td>
</tr>
<tr>
<td>Properties file</td>
<td></td>
</tr>
<tr>
<td>JSON file (custom)</td>
<td></td>
</tr>
<tr>
<td>Vertical File</td>
<td>Retrieve text from a structured text file where each set of data spans</td>
</tr>
<tr>
<td></td>
<td>multiple lines. For more information, see Parse text from a vertical file.</td>
</tr>
<tr>
<td>After Keyword</td>
<td>Retrieve text directly following a specific keyword. For more information,</td>
</tr>
<tr>
<td></td>
<td>see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Command Line Java Style</td>
<td>Retrieve the value of a command-line parameter using Java-style parameters.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Parse text using keyword, command, and positional</td>
</tr>
<tr>
<td>Command Line Unix Style</td>
<td>type.</td>
</tr>
<tr>
<td></td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Parse text using keyword, command, and positional</td>
</tr>
<tr>
<td></td>
<td>type.</td>
</tr>
<tr>
<td>Position From End</td>
<td>Retrieve text specified by its position from the end of the line. For</td>
</tr>
<tr>
<td></td>
<td>more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Position From Start</td>
<td>Retrieve text specified by its position from the beginning of the line. For</td>
</tr>
<tr>
<td></td>
<td>more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Retrieve text specified by a regular expression. This option requires</td>
</tr>
<tr>
<td></td>
<td>familiarity with Regex Java syntax. For more information, see Parse text</td>
</tr>
<tr>
<td></td>
<td>using a regular expression.</td>
</tr>
<tr>
<td>Delimited Text</td>
<td>Retrieve text specified by delimiters and position within the line (the</td>
</tr>
<tr>
<td></td>
<td>most common way to retrieve text from generic text files). See Parse text</td>
</tr>
<tr>
<td></td>
<td>using delimited text.</td>
</tr>
</tbody>
</table>

4. Select Terminate to stop discovery if no results are found.
5. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

You can use the Command Line Java Style parsing strategy to extract the path of the installation directory of the WebSphere Server.
• Continue editing the pattern by adding a new step and defining its operation or Finalize the pattern.

Define an HTTP Get Call query
As part of creating or modifying a discovery pattern, you can use the **Http Get Call** operation to extract information from configuration items (CIs), which use the HTTP protocol.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin

1. Select **Http Get Call** from the **Operation** list.
2. Define query parameters as necessary:

   ![Warning] If you customized this operation as described in [Customize pattern operations](#), the query parameters may be different.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required Authentication</td>
<td>If authentication is required for discovering the CI:</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Set this parameter to <strong>true</strong>.</td>
</tr>
<tr>
<td></td>
<td>· Configure credentials for the CI you want to discover as described in <a href="#">Basic authentication credentials</a>.</td>
</tr>
<tr>
<td></td>
<td>Set this parameter to <strong>false</strong> or leave empty if no authentication is required for accessing the CI.</td>
</tr>
<tr>
<td>URL</td>
<td>Specify the URL of the CI to discover using this pattern.</td>
</tr>
<tr>
<td></td>
<td>You can use variables. You can also enter a value from the specific field in a tabular variable as described in <a href="#">Enter values and variables in patterns</a>.</td>
</tr>
<tr>
<td>Headers</td>
<td>(Optional) Define HTTP headers to define the output format. For example, to see the output in the JSON format and bring in the security policy data, configure the header as follows:</td>
</tr>
<tr>
<td></td>
<td><code>content-type:text/JSON,</code></td>
</tr>
</tbody>
</table>

3. To save the command output in its entirety as a variable, select **NONE** from the **Define Parsing** list and define the variable as described in step 6.
4. Select the parsing strategy from the **Define Parsing** list.
<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>Horizontal file parsing strategy (not vertical). You can use this parsing strategy only for text files. For more information, see <a href="#">Parse text from a horizontal file</a>.</td>
</tr>
<tr>
<td>LDAP file</td>
<td></td>
</tr>
<tr>
<td>XML file</td>
<td></td>
</tr>
<tr>
<td>INI file</td>
<td></td>
</tr>
<tr>
<td>Properties file</td>
<td></td>
</tr>
<tr>
<td>JSON file (custom)</td>
<td></td>
</tr>
<tr>
<td>Vertical File</td>
<td>Retrieve text from a structured text file where each set of data spans multiple lines. For more information, see <a href="#">Parse text from a vertical file</a>.</td>
</tr>
<tr>
<td>After Keyword</td>
<td>Retrieve text directly following a specific keyword. For more information, see <a href="#">Parse text using keyword, command, and positional type</a>.</td>
</tr>
<tr>
<td>Command Line Java Style</td>
<td>Retrieve the value of a command-line parameter using Java-style parameters. For more information, see <a href="#">Parse text using keyword, command, and positional type</a>.</td>
</tr>
<tr>
<td>Command Line Unix Style</td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters. For more information, see <a href="#">Parse text using keyword, command, and positional type</a>.</td>
</tr>
<tr>
<td>Position From End</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see <a href="#">Parse text using keyword, command, and positional type</a>.</td>
</tr>
<tr>
<td>Position From Start</td>
<td>Retrieve text specified by its position from the beginning of the line. For more information, see <a href="#">Parse text using keyword, command, and positional type</a>.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Retrieve text specified by a regular expression. This option requires familiarity with Regex Java syntax. For more information, see <a href="#">Parse text using a regular expression</a>.</td>
</tr>
<tr>
<td>Delimited Text</td>
<td>Retrieve text specified by delimiters and position within the line (the most common way to retrieve text from generic text files). See <a href="#">Parse text using delimited text</a>.</td>
</tr>
</tbody>
</table>

5. If working in the Debug Mode, define the parsing criteria as follows:
   a) Click **Run Operation** to see the result in the **Output** pane.
   b) In the **Output** pane, mark text or symbols that you want a variable to contain.
   c) In the variable name box, enter the name for the new variable, for example ‘process\_name’.
6. If you are not working in Debug Mode, define the parsing criteria as follows:
   a) In the Variables pane, click **Add Variable** and enter the name for the new variable.
   b) Click the **Advanced** icon.
   c) Click **Add Column Label**.

   d) Press **Enter**.

   The new variable is added in the **Variables** pane.
d) Enter the parsing query.

e) Enter a value for the delimiter.

f) Enter a value for the position.

7. Select **Terminate** to stop discovery if no results are found.

8. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

You can use the HTTP Get Call operation to extract data on CI types using the XML File parsing strategy:

- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.

Define an API Query for cloud computing devices

As part of creating or modifying a discovery pattern, you can use the **Cloud REST Call** operation to extract information from configuration items of the PaaS (Platform as a Service) type, such as Microsoft Azure or Amazon Web Services.

- Verify that the operating system of the configuration item (CI) for which you want to use the Cloud REST Call operation extends the Logical Data Center OS type (**cmdb_ci_logical_datacenter**):
1. Navigate to the Basic tab of the CI pattern.
2. Note the operating system for this CI.
3. Navigate to System Definition > Tables.
4. Set the search field to Label and enter the name of the operating system as stated on the Basic tab of the pattern.
5. Find the operating system in the list and verify that Logical Datacenter appears in the Extends table column.

- Navigate to the relevant pattern step:
  1. On the pattern form, select the relevant identification section for Discovery.
     Alternatively, select the relevant identification or connection section for Service Mapping.
  2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

Deploy the Cloud REST Call operation in patterns used for discovery of PaaS CIs.

Refer to the official API-related documentation provided by manufacturers to obtain the query syntax for the device you want to query using the Cloud REST Call operation. For example:

- [https://docs.microsoft.com/en-us/rest/api/apimanagement/](https://docs.microsoft.com/en-us/rest/api/apimanagement/)

1. Select Cloud REST Call from the Operation list.
2. Define query parameters as necessary:

   **Warning:** If you customized this operation as described in Customize pattern operations, the query parameters may be different.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td>Specify the URL as described in the official Microsoft Azure or Amazon Web Services documentation. You can use variables. You can use variables. You can also enter a value from the specific field in a tabular variable as described in Enter values and variables in patterns.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Method | Enter the method value as a string using all capital letters, for example “GET”. The supported methods are:
- GET
- POST
- PUT
Refer to the relevant API guide for information. If the API documentation does specifically mention the HTTP query method, use the GET method.

Body | (Optional) Enter a request body as a string. If the string contains quotation marks, use the backslash mark in front the quotation marks to indicate that the string does not end at the quotation marks.

Headers | (Optional) If the relevant API documentation states that HTTP headers must be sent, enter these headers in the following format:
```
Header_name1:header_value1,header_name2:header_value2
```
For example, `Content-Type:xml`.

3. To save the command output in its entirety as a variable, select **NONE** from the **Define Parsing** list and define the variable as described in 6.

4. Select the parsing strategy from the **Define Parsing** list.

| Parsing strategy | Description |
--- | ---
Oracle | Horizontal file parsing strategy (not vertical). You can use this parsing strategy only for text files. For more information, see [Parse text from a horizontal file](#).
LDAP file | 
XML file | 
INI file | 
Properties file | 
JSON file (custom) | 
Vertical File | Retrieve text from a structured text file where each set of data spans multiple lines. For more information, see [Parse text from a vertical file](#).
After Keyword | Retrieve text directly following a specific keyword. For more information, see [Parse text using keyword, command, and positional type](#).
Command Line Java Style | Retrieve the value of a command-line parameter using Java-style parameters. For more information, see [Parse text using keyword, command, and positional type](#).
<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Line Unix Style</td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Position From End</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Position From Start</td>
<td>Retrieve text specified by its position from the beginning of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Retrieve text specified by a regular expression. This option requires familiarity with Regex Java syntax. For more information, see Parse text using a regular expression.</td>
</tr>
<tr>
<td>Delimited Text</td>
<td>Retrieve text specified by delimiters and position within the line (the most common way to retrieve text from generic text files). See Parse text using delimited text.</td>
</tr>
</tbody>
</table>

5. If working in the Debug Mode, define the parsing criteria as follows:
   a) Click **Run Operation** to see the result in the **Output** pane.
   b) In the **Output** pane, mark text or symbols that you want a variable to contain.
   c) In the variable name box, enter the name for the new variable, for example ‘process_name’.
   d) Press **Enter**.
      The new variable is added in the **Variables** pane.

6. If you are not working in Debug Mode, define the parsing criteria as follows:
   a) In the Variables pane, click **Add Variable** and enter the name for the new variable.
b) Click the **Advanced** icon.

c) Click **Add Column Label**.

d) Enter the parsing query.

e) Enter a value for the delimiter.

f) Enter a value for the position.

7. Select **Terminate** to stop discovery if no results are found.

8. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Application</td>
</tr>
<tr>
<td>CI Type</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Amazon AWS Elastic Load Balancer Service</td>
</tr>
<tr>
<td>Section</td>
<td>AWS Load Balancer Service</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>2. Get first 100 RDS instances</td>
</tr>
</tbody>
</table>

Use the Cloud REST Call operation to extract information on Database instances running on the region us-west-2 in the Amazon cloud.
Continue editing the pattern by adding a new step and defining its operation or Finalize the pattern.

Create a relationship and a reference
As part of creating or modifying a discovery pattern, you can use the Relation and/or Reference condition to create relationships and references between CIs a pattern discovers. This operation is relevant for both infrastructure and application patterns that Discovery uses for horizontal discovery.

Decide on the nature of relationship or reference that you are going to create.

The CI relationship you define in the pattern must comply with the model you have created:

1. Check if the CI is dependent: The system uses the identification rules to identify the CI independently of other CIs. Refer to Create or edit a CI identification rule.

2. If the CI is dependent, check its dependent relationship rules which define dependency structure of the CI types and the relationship types in these service definitions. For example, you can discover a load balancer pool with pool members belonging to it. To correctly reflect
the type of relationship between these two CIs, create a containment rule, which defines the load balancer pool as the owner of pool members: **Runs on::Runs**. For more information, refer to **CMDB dependent relationship rules**.

Example of a dependent rule for Load Balancer Pool Member

For independent CI types without dependent relationship rules define for them, you can define any relationship.

3. (Optional for application patterns used for horizontal discovery) Make sure that the data discovered using this pattern is saved as a table using standard parsing operations.
4. (Optional for creating a reference) Navigate to the relevant table definitions for the parent and the child CIs and choose the field to use as a reference field.

5. Navigate to the relevant pattern step:
   a. On the pattern form, select the relevant identification section.
   b. Select the relevant step or create a pattern step by clicking .

Basic knowledge of programming is desirable.
Role required: pd_admin

Discovery uses some patterns to discover a CI with all its related CIs and non-CIs: items that do not extend the Configuration Item (cmdb_ci) table.

CIs can have different relationships. Dependent relationship rules describe relationships between CIs. The identification engine uses these Dependent relationship rules to uniquely identify CIs and determine if a specific CI exists in the CMDB or must be added to the CMDB.

When you create a relationship, the system uses the parent CI and the child CI tables and creates a third table with data on the relationship between the parent and child CIs.

In addition to a relationship between CIs, you can create a reference connection between them. If a reference exists, then information about referenced child CIs appears at the bottom of the parent CI form. For example, all load balancer pool members appear on the form of the load balancer pool, which owns these members.
1. Discover related items together with the main CI to be able to view and use attributes of related CI types.
2. Select Create Relation/Reference from the Operation list.
3. Add the related CI type to this pattern as described in Discover related items together with the main CI.
4. Complete the form using the fields in the table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Table</td>
<td>Enter the name of the table for the CI that you want to use as a parent CI in the relation. For example, cmdb_ci_lb_pool for the load balancer pool CI.</td>
</tr>
<tr>
<td>Child Table</td>
<td>Enter the name of the table for the CI that you want to use as a child CI in the relation. For example, cmdb_ci_lb_pool_member for the load balancer pool member, which the load balancer pool owns.</td>
</tr>
<tr>
<td>Result Table</td>
<td>Enter the name for the new table to store information about relations and references between CIs, resulting from this operation.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Relation Type         | Select the type of relationship between CIs from the specified target tables. The part in the type name before the separator (::) refers to the parent CI and the second part, after the separator (::), to the child CI. For example, in the Owns::Owned by option:  
  - ‘Owns’ indicates that a load balancer pool (cmdb_ci_lb_pool) owns a load balancer pool member (cmdb_ci_lb_pool_member).  
  - ‘Owned by’ indicates that a load balancer pool member (cmdb_ci_lb_pool_member) is owned by a load balancer pool (cmdb_ci_lb_pool). |
| Reference              | (Optional) Select this check box to create a reference between the parent and the child CIs. |
| Direction              | (Only if Reference is selected) Select the direction of the reference:  
  - **Parent to Child** to display the parent CI on the child CI form. For example, the load balancer pool member form displays load balancer pool, which owns this pool member.  
  - **Child to Parent** to display child CIs on the parent CI form. For example, the load balancer pool form displays load balancer pool members this pool owns. |
<p>| Column Name            | (Only if Reference is selected) Enter the name of the field in the child CI table that the system uses to create the reference. For example, load_balancer. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation Criteria</td>
<td>Select the criteria used to create the relation and/or reference:</td>
</tr>
<tr>
<td></td>
<td><strong>All</strong></td>
</tr>
<tr>
<td></td>
<td>The system creates a relationship or reference without any additional criteria.</td>
</tr>
<tr>
<td></td>
<td><strong>Field matching</strong></td>
</tr>
<tr>
<td></td>
<td>The system creates the relationship or reference only if the value in the specified field in a parent CI matches the value of the specified field in a child CI.</td>
</tr>
<tr>
<td></td>
<td><strong>Condition</strong></td>
</tr>
<tr>
<td></td>
<td>The system creates the relationship or reference only if the specified criteria is matched.</td>
</tr>
</tbody>
</table>

Parent table field

(Only if Field matching is selected) Enter the field from the parent table to compare.

You can use variables including values from tabular variables as described in Enter values and variables in patterns.

Child table field

(Only if Field matching is selected) Enter the field from the child table to compare.

Unmatched values

Operation for values in the two tables that do not match. You can keep or remove unmatched values.

5. If you selected **Condition** from **Creation Criteria**, define this condition:
   a) In the first condition field, enter the required value.
   b) In the first condition field, enter the required value.
      For example, enter the actual string or a variable name.

   ![Set values](image.png)

   ![Add another condition](image.png)

   ![Select an operator](image.png)

   c) Select an operator from the list.
If you select Is Empty, the second field is rendered irrelevant and disappears.

d) To add more conditions, click the plus icon and define the criteria.

e) If you create multiple conditions, define if this pattern must meet all or any of them: From the Meet list, select All or Any.

Use the **Create Relation/Reference** operation to create the Owns::Owned by relation between a load balancer pool and a load balancer pool member. You also create a reference from a load balancer pool member to the load balancer pool.
Another example is using the **Create Relation/Reference** operation to create the **Provides::Provided by** relation between a storage device and a file system. This operation also creates a reference from the storage device to the load system.
Another example is using the **Create Relation/Reference** operation to create the reference between a disk partition and a iSCSI disk.
The example below shows how to add a non-CI item (a switch port (dscy_switchport)) to a CI (switch (cmdb_ci_ip_switch)).

```
```
```
- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.
Reuse a shared step library
As part of creating or modifying a discovery pattern, you can reuse a sequence of discovery steps that you created for one pattern in other patterns. Saved shared step sequences become pattern modules referred to as shared libraries.

Basic knowledge of programming is desirable.

Role required: pd_admin

Reusing existing step sequences saves you from recreating the same steps manually.

First, you save the steps you want to reuse into a repository as a shared library. Then you can insert them wherever relevant and as many times as needed.

You can deactivate shared libraries, so that patterns using them, skip steps in deactivated shared libraries. For more information, see *Discovery Configuration Console*.

1. If the steps you want to reuse already exist, create a reusable step sequence as follows:
   a) Navigate to the pattern containing the steps that you want to reuse.
   b) Navigate to the steps that you want to combine.
   c) Select the steps, right-click, and choose *Create shared library*.
   d) Fill in the fields as appropriate:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shared Library Name</td>
<td>Specify a unique name for the library. For example, UNIX-OS for a step sequence used to discover the Unix operating system.</td>
</tr>
<tr>
<td>Library Description</td>
<td>Specify a meaningful description for the library.</td>
</tr>
</tbody>
</table>

   e) Click *Create*.
   The selected steps are saved as a shared library in the Discovery Patterns (sa_pattern) table.

2. Reuse the step sequence you created:
   a) Navigate to the pattern in which you want to insert your step sequence.
   b) Select the relevant identification section or connectivity section.
   c) Select *Library Reference* from the *Operation* list in one of the following locations:
      - the *Identification Sections* or *Connectivity Sections* for Service Mapping
      - the Step window for Discovery
   d) Select the required shared library from the *Library* list.
   The library is inserted as substeps inside an untitled step.
e) Rename **Untitled Step** to reflect the purpose of the step.

3. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

- Continue editing the pattern by **adding a new step and defining its operation** or
- **Finalize the pattern**.

**Run an SSH script file**

As part of creating or modifying a discovery pattern, you can use the **Run SSH Script File** operation to run composite commands or sequences of commands on Unix-based hosts.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. 

   Select the relevant pattern step or click ![icon](image) to add a step.

**Basic knowledge of programming is desirable.**

**Role required:** pd_admin

1. Select **Run SSH Script File** from the **Operation** list.

2. Enter file name of the SSH script file located on the MID Server in the **File Name**.

3. To save the command output in its entirety as a variable, select **NONE** from the **Define Parsing** list and define the variable as described in step 6.

4. If necessary, select the parsing strategy from the **Define Parsing** list.

**Note:** Alternatively, use EVAL scripts to manipulate collected data as described in **KB0647736: Examples of EVAL scripts used in discovery patterns**.
<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>Horizontal file parsing strategy (not vertical). You can use this parsing strategy only for text files. For more information, see Parse text from a horizontal file.</td>
</tr>
<tr>
<td>LDAP file</td>
<td></td>
</tr>
<tr>
<td>XML file</td>
<td></td>
</tr>
<tr>
<td>INI file</td>
<td></td>
</tr>
<tr>
<td>Properties file</td>
<td></td>
</tr>
<tr>
<td>JSON file (custom)</td>
<td></td>
</tr>
<tr>
<td>Vertical File</td>
<td>Retrieve text from a structured text file where each set of data spans multiple lines. For more information, see Parse text from a vertical file.</td>
</tr>
<tr>
<td>After Keyword</td>
<td>Retrieve text directly following a specific keyword. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Command Line Java Style</td>
<td>Retrieve the value of a command-line parameter using Java-style parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Command Line Unix Style</td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Position From End</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Position From Start</td>
<td>Retrieve text specified by its position from the beginning of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Retrieve text specified by a regular expression. This option requires familiarity with Regex Java syntax. For more information, see Parse text using a regular expression.</td>
</tr>
<tr>
<td>Delimited Text</td>
<td>Retrieve text specified by delimiters and position within the line (the most common way to retrieve text from generic text files). See Parse text using delimited text.</td>
</tr>
</tbody>
</table>

5. If working in the Debug Mode, define the parsing criteria as follows:
   a) Click Run Operation to see the result in the Output pane.
   b) In the Output pane, mark text or symbols that you want a variable to contain.
   c) In the variable name box, enter the name for the new variable, for example ‘process_name’.
d) Press **Enter**.  
The new variable is added in the **Variables** pane.

6. If you are not working in Debug Mode, define the parsing criteria as follows:  
a) In the Variables pane, click **Add Variable** and enter the name for the new variable.  
b) Click the **Advanced** icon.

c) Click **Add Column Label**.
d) Enter the parsing query.
e) Enter a value for the delimiter.
f) Enter a value for the position.

7. Select **Terminate** to stop discovery if no results are found.
8. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Hardware &gt; Computer &gt; Server &gt; Linux Server</td>
</tr>
<tr>
<td>CI Type</td>
<td>Linux Server (cmdb_ci_linux_server)</td>
</tr>
<tr>
<td>Pattern</td>
<td>Linux Server</td>
</tr>
<tr>
<td>Section</td>
<td>discovery</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>13.1 Run storage script file</td>
</tr>
</tbody>
</table>

This pattern step uses the Run SSH Script File operation to collect all information about storage entities associated with the discovered Linux server.
- Continue editing the pattern by adding a new step and defining its operation or Finalize the pattern.

Set a parameter value
As part of creating or modifying a discovery pattern, you can use the Set Parameter Value operation to apply a value to a parameter.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin

1. On the Identification or Connectivity Sections form, select **Set Parameter Value** from the **Operation** list.
2. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Value</strong></td>
<td>Specify the value that you want to assign to the parameter. You can use strings or variables. You can also use values from temporary tabular variables: from a specific field or a specific column in a table sequentially, starting from the first row. For more information, see <a href="#">Enter values and variables in patterns</a>.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Specify the name of the variable to which to assign this value.</td>
</tr>
</tbody>
</table>

3. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Application</td>
</tr>
<tr>
<td>CI Type</td>
<td>Microsoft iis Web Server (cmdb_ci_microsoft_iis_web_server)</td>
</tr>
<tr>
<td>Pattern</td>
<td>IIS</td>
</tr>
<tr>
<td>Section</td>
<td>Identification for JHTTP(s) entry point.</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>3. Set the display label</td>
</tr>
</tbody>
</table>

To set the CI name to “IIS” and the version number, define the **Set Parameter Value** operation as follows:
Transfer a file
As part of creating or modifying a discovery pattern, you can transfer a file from the MID Server to a computer in your organization.

Basic knowledge of programming is desirable.

Role required: pd_admin

1. Upload the file to the MID Server:
   a) Navigate to Pattern Designer > Uploaded Files.
   b) Click New.
   c) Enter a meaningful name in the Name field.
   d) Click the Operating Systems tab.
   e) Click OS Types and select the relevant type from the list.
f) Click **OS Architectures** and select the relevant option from the list.

   **Note:** You can select both 32-bit and 64-bit option for the same file if necessary.

   g) Click the **Manage Attachments** icon.

   h) Attach a file.

   See **Add and manage attachments**.

2. Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

3. Select **Put file** from the **Operation** list.

4. Click **Refresh**.
   
The name of the uploaded file appears in the **File name** field.

5. Specify the variable that refers to the MID Server path on the remote computer in the **Full Path Target** field.

6. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Application</td>
</tr>
<tr>
<td>CI Type</td>
<td>ActiveMatrix Business Works (cmdb_ci_appl_tibco_matrix)</td>
</tr>
<tr>
<td>Pattern</td>
<td>ActiveMatrix Business Works</td>
</tr>
<tr>
<td>Section</td>
<td>Tibco BW Identification</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>20. put tibco script</td>
</tr>
</tbody>
</table>

Use the Put File operation to copy the TibcoParser file onto the target host and keep the full path to copied file in target host in the variable `$TibcoParser`. 
Continue editing the pattern by adding a new step and defining its operation or finalize the pattern.

Define an SNMP query
As part of creating or modifying a discovery pattern, you can use the SNMP query operation to execute SNMP queries on CIs that support the SNMP protocol, typically, load balancers.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery. Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin
1. Select SNMP query from the Operation list.
2. To browse to the SNMP OID that you want to query using the debug mode:
   a) Activate pattern Debug mode.
b) Click **Browse** and select the management information base (MIB) in the list or perform a search based on MIB values.

c) Select either a single scalar value or multiple columns in the MIB tree. Multiple columns must be in the same table.

d) Optional: Click **Get Data** to display the data in the SNMP Browser form.

e) Click **OK**.

3. To manually specify the SNMP OIDs for the query:

- If the result of the SNMP query is a single string, choose **Scalar** and fill in the fields:

  **Scalar fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP OID</td>
<td>The SNMP OID to query.</td>
</tr>
<tr>
<td>Variable</td>
<td>Name of the scalar variable to hold the query result.</td>
</tr>
</tbody>
</table>

- If the result of the SNMP query is a table, choose **Table**, click the plus icon, and fill in the fields:

  **Table fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table name</td>
<td>Replace <em>Table_Name</em> with the name of the Variables table.</td>
</tr>
<tr>
<td>Table OID</td>
<td>Replace <em>Table_OID</em> with the SNMP OID to query.</td>
</tr>
<tr>
<td>New column name</td>
<td>Replace <em>new-column-name</em> with the column name.</td>
</tr>
<tr>
<td>New OID</td>
<td>Replace <em>new_OID</em> with the SNMP OID to query.</td>
</tr>
</tbody>
</table>

4. Select **Use Cache** to save the operation results in cache on the MID Server.

Use cache to optimize discovery and avoid creating unnecessary load on central shared components, such as load balancers. The base system keeps operation results in cache for an hour.

5. Select **Terminate** to stop discovery if no results are found.

6. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

The following configuration is an example of the **SNMP query** operation.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Network &gt; Load Balancer</td>
</tr>
<tr>
<td>CI Type</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
</tr>
<tr>
<td>Pattern</td>
<td>F5 BigIP LTM</td>
</tr>
<tr>
<td>Section</td>
<td>Identification for all entry point types</td>
</tr>
</tbody>
</table>
The F5 BigIP LTM pattern uses the SNMP Query operation to extract the product name of the load balancer.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step number and Name</td>
<td>1. Get Name (BIG IP LB)</td>
</tr>
</tbody>
</table>

- Continue editing the pattern by **adding a new step and defining its operation** or **Finalize the pattern**.

**Add a column to a table**
As part of creating or modifying a discovery pattern, you can use the **Transform table** operation to add one or more computed columns to an existing table and place the results in a target table. The target table can be the source table.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
1. Select **Transform table** from the **Operation**.
2. Fill in the fields, as appropriate.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source table</td>
<td>Specify the name of the source table.</td>
</tr>
<tr>
<td>Target table</td>
<td>Specify the name of the target table. The target table can be the same as</td>
</tr>
<tr>
<td></td>
<td>the source table.</td>
</tr>
</tbody>
</table>

3. Click the plus icon to add each target field and fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Field Name</td>
<td>Specify a name for the column. It can be an existing column or a new one.</td>
</tr>
<tr>
<td>Value</td>
<td>Specify the operation expression that determines the values added to the</td>
</tr>
<tr>
<td></td>
<td>column. You can use variables including values from tabular variables as</td>
</tr>
<tr>
<td></td>
<td>described in Enter values and variables in patterns.</td>
</tr>
</tbody>
</table>

4. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

Identity Integration Feature Pack

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Web Servers</td>
</tr>
<tr>
<td>CI Type</td>
<td>Load Balancer Service (cmdb_ci_lb_service)</td>
</tr>
<tr>
<td>Pattern</td>
<td>F5 BigIP LTM</td>
</tr>
<tr>
<td>Section</td>
<td>Identification for all entry point types</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>9. Fix ip</td>
</tr>
</tbody>
</table>

For F5 load balancer, you want to translate IP addresses in the default hexadecimal format into decimal format. You use the Javascript that performs this translation in the value field. You add the "clean_ip" column containing IP addresses in decimal format to the existing (source) table.
Continue editing the pattern by adding a new step and defining its operation or finalize the pattern.

Change credentials to default
As part of creating or modifying a discovery pattern, you can use the Unchange user operation to switch back to default administrative credentials if you previously changed them to non-default using the Change user operation.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin
1. Select Unchange User from the Operation list.
2. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Business Integration Software</td>
</tr>
<tr>
<td>CI Type</td>
<td>IBM WebSphere MQ Queue</td>
</tr>
<tr>
<td>Pattern</td>
<td>WMQ Queue Unix Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Local queue connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>11. exit change user credentials</td>
</tr>
</tbody>
</table>

As part of discovery of an IBM WebSphere MQ Queue, you change credentials to IBM WebSphere MQ. It allows you to manipulate local IPs to discover connections on the local queue. Once this segment of discovery is complete, you change credentials back to the default: IBM WebSphere MQ Queue credentials.
• Continue editing the pattern by **adding a new step and defining its operation** or **Finalize the pattern**.

**Append two tables**
As part of creating or modifying a discovery pattern, you can use the **Union table** operation to append two tables of the same format.

Navigate to the relevant pattern step:
1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click ![Add Step](image) to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

Appending two tables adds the second table at the end of the first table and places the results in the additional, target table. If necessary, you can specify one of the source tables as the target table.

1. Select **Union Tables** from the **Operation** list.
2. Fill in the fields, as appropriate.
## Field | Description
--- | ---
First table | Specify the name of the first source table.
Second table | Specify the name of the second source table.
Target table | Specify the name of the table into which the results are placed.

3. If in Debug mode, test the step by clicking **Test** and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Application Servers</td>
</tr>
<tr>
<td>CI Type</td>
<td>WebSphere EAR</td>
</tr>
<tr>
<td>Pattern</td>
<td>J2EE EAR on Linux</td>
</tr>
<tr>
<td>Section</td>
<td>Oracle JDBC connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>5. Union JDBCProviders with JDBCProviders2</td>
</tr>
</tbody>
</table>

A WebSphere EAR on Linux has two tables containing data on JDBC Providers. To add data from the second table at the bottom of the first table, use the **Union Tables** operation as follows:

- Continue editing the pattern by **adding a new step and defining its operation** or **Finalize the pattern**.

**Define WMI method invocation**

As part of creating or modifying a discovery pattern, you can use the **WMI method invocation** operation to execute a method selected from a table returned by a WMI query.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Make sure that the step containing the WMI method invocation operation follows the step with the WMI query operation. The WMI query results in table which you must use as the source table for WMI method invocation operation.

Basic knowledge of programming is desirable.

Role required: pd_admin

This operation is relevant only for Windows.

1. Select WMI method invocation from the Operation list.
2. Fill in the fields, as appropriate:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter Source Table</td>
<td>Specify the source table name. The source table must be the result of the WMI Query operation you perform before this step. You can enter a value from the specific field in a table as described in Enter values and variables in patterns.</td>
</tr>
</tbody>
</table>
| Enter WMI Method     | Select the desired method:
|                      |  - In the debug mode, click Get methods and select the method from the list. |
|                      |  - Not in the debug mode, enter the name of the method as a string, for example, “RunDetails”. |
| Enter Target Table   | Specify the target table name.                                              |
| Enter Target Column  | Specify the name of the column to contain the results of the method invocation. |

3. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Directory Services</td>
</tr>
<tr>
<td>CI Type</td>
<td>IIFP</td>
</tr>
<tr>
<td>Pattern</td>
<td>IIFP On Windows Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>AD Home Forest connectivity stage-wmi</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>2. invoke_wmi_method Run Details</td>
</tr>
</tbody>
</table>

During discovery of Identity Integration Feature Pack (IIFP), use the RunDetails WMI method to extract information on details from the ManagementAgents table. You discover this table earlier using the WMI query operation. In this case, the result is saved in the same table in the column named “details”.
- Continue editing the pattern by **adding a new step and defining its operation** or
- **Finalize the pattern**.

**Define a WMI query**

As part of creating or modifying a discovery pattern, you can use the **WMI query** operation to execute a query on a remote Windows system either explicitly or automatically. Successful values are logged in a specified target table.

Navigate to the relevant pattern step:

1. On the pattern form, select the relevant identification section for Discovery.
   Alternatively, select the relevant identification or connection section for Service Mapping.

2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.

Role required: pd_admin

This operation is relevant only for Windows.

1. Select **WMI Query** from the **Operation** list.
2. Define the namespace path in the **Namespace** field.
3. If in the debug mode:
   a) Click **Statement Accelerator** to populate the table and field values automatically.
   b) Select a table from the **Table name** pane.
      The field values are displayed in the **Fields** pane.
   c) Define parameters and operators for the condition in the **Add Condition Clause**.
      The actual query text is displayed in the **Get Full Statement**.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
4. If not in the debug mode, enter the WMI query string in the Get Full Statement.
5. Enter the variable for the table in which the retrieved data is saved in the Add Target Table field.
   You can enter a value from the specific field in a table as described in Enter values and variables in patterns.
6. Select Use Cache to save the operation results in cache on the MID Server.
   Use cache to optimize discovery and avoid creating unnecessary load on central shared components, such as load balancers. The base system keeps operation results in cache for an hour.
7. Select Terminate to stop discovery if no results are found.
8. If in Debug mode, test the step by clicking Test and checking that the operation brings the result you expected.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Directory Services</td>
</tr>
<tr>
<td>CI Type</td>
<td>IIFP (cmdb_ci_directory_iifp)</td>
</tr>
<tr>
<td>Pattern</td>
<td>IIFP On Windows Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>AD Home Forest connectivity stage-wmi</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>1. Get all agents details via WMI step</td>
</tr>
</tbody>
</table>

This step uses the WMI Query operation to extract data on agents from the MicrosoftIdentityIntegrationAgent namespace and save this data in the variable table named $ManagementAgents.

- Continue editing the pattern by adding a new step and defining its operation or
- Finalize the pattern.
Parsing strategies
In discovery patterns, you can use parsing strategies to analyze syntax of the source file. You extract values from parsed files, which allows you later to convert these values into variables.

There are several parsing strategies coming with the base system:

<table>
<thead>
<tr>
<th>Parsing strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>Horizontal file parsing strategy (not vertical). You can use this parsing strategy only for text files. For more information, see Parse text from a horizontal file.</td>
</tr>
<tr>
<td>LDAP file</td>
<td>Retrieve text from a structured text file where each set of data spans multiple lines. For more information, see Parse text from a vertical file.</td>
</tr>
<tr>
<td>XML file</td>
<td>Retrieve text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>INI file</td>
<td>Retrieve the value of a command-line parameter using standard Unix parameters. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>Properties file</td>
<td>Retrieve text specified by its position from the end of the line. For more information, see Parse text using keyword, command, and positional type.</td>
</tr>
<tr>
<td>JSON file (custom)</td>
<td>Retrieve text specified by delimiters and position within the line (the most common way to retrieve text from generic text files). See Parse text using delimited text.</td>
</tr>
</tbody>
</table>

In addition to these parsing strategies, you can create custom parsing strategies to answer the needs of your organization.

Parse text from a horizontal file
You can use the file type parsing strategy to parse text in files of the following formats: .ora file (used by various Oracle products), .properties file (common for Java), .xml file, and .ini file. For vertical files, use the vertical file parsing strategy instead.
Role required: pd_admin

Basic knowledge of programming is desirable.

You can use this parsing strategy only for text files.

⚠️ Warning: Do not use this parsing strategy for non-text files such as binary files.

You can define multiple extracts and variables. When identifying text for extraction into variables, what you are really doing is identifying the text location within a context.

You can use one of the following methods:

- In Debug mode, you can select the relevant string from the file contents in the text box. For each string you select, its position and delimiters relative to its context are stored. It enables the same definitions to apply to other files with the same structure even though the text varies. However, it selects the entire text within a context.

  For example, if you try to select only 456 in the text box of an XML file with the following line, the entire string between the keywords is selected.

  ```xml
  <ciTypeID>123-456-7890000000</ciTypeID>
  ```

- On the Advanced Parsing Options form (outside of Debug mode), you can specify a delimiter and position to identify the text string. You can also use this form to make a more refined selection than from within the text box.

  For example, you could specify a delimiter (→) and the number of positions to extract after the delimiter (3) to extract the string (456).

1. Select one of the parsing operations from the Operation list.
2. Select one of these options from the Parsing strategy list:
   - Oracle
   - JSON file
   - Properties file
   - XML file
   - LDAP file
   -INI file

3. If working in the Debug mode, perform the following steps:
   a) Click Retrieve or Retrieve File Content to display the content of what you are parsing in the Output pane.
   b) Select the string in the text box. All matching strings in the same context are automatically selected.
   c) Assign the string to a variable on the Define Variable Name form. Provide a unique and meaningful name and click OK.
   d) To identify additional strings and variables, click the plus icon.

4. Define the string to be parsed from within Debug mode, or on the Advanced Parsing Options form (outside of Debug mode).
2. Define Parsing

- XML File

Output

<table>
<thead>
<tr>
<th>Variables</th>
<th>Table</th>
<th>List</th>
<th>Add Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>There must be at least one entry</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Advanced Parsing Options

- Root Path

<table>
<thead>
<tr>
<th>Name</th>
<th>XPath query</th>
<th>Delimiter</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Advanced Parsing Options form

1. Click Advanced and specify the root path. The root path is the section (hierarchical branch in the file structure) where parsing takes place.
2. Click the plus icon for each string and variable to be added and fill in the fields, as appropriate.
   - **Name**: Specify the column name.
   - **XPath query**: Specify the XPath query for the string. For example, appcmd/APP/@APP.NAME.
   - **Delimiter**: Specify the delimiter for the string.
   - **Position**: Specify the position of the string.

5. To end the discovery process if no results are found, select the **If not found** check box.
6. Click **Close Advanced**.

Parse text from a vertical file

In discovery patterns, you can use the **Vertical file** parsing strategy to parse text into variables for vertical files.

Basic knowledge of programming is desirable.

Role required: pd_admin

1. Select one of parsing operations from the **Operation** list in one of the following locations:
- the **Identification Sections** or **Connectivity Sections** for Service Mapping
- the Step window for Discovery

2. Select the **Vertical file** from the **Parsing strategy** list.
3. In the **Target Table** field, enter the name of the table to hold variables extracted during this operation.
4. In the **Section** field, enter the title of the section.
   This parameter is used to identify the beginning of sections in the file.
5. In the **Line** field, enter the line number. The lines are counted from top to bottom.
6. In the **Column** field, enter the column number. The columns are counted from left to right.
7. To end the discovery process if no results are found, select the **If not found** check box.

You can use the **Vertical File** parsing strategy for the Parse Command Output operation to extract information about a database. The section separator is `entry:`.
Parse text using keyword, command, and positional type

In discovery patterns, you can parse text into a variable for keyword, command, and positional type parsing strategies.

Basic knowledge of programming is desirable.

Role required: pd_admin

The following strategies are generally used to extract a value from a variable. You can use them to define only a single value for extraction. To extract multiple strings, define multiple steps.

- After keyword
- Command-line Java style
- Command-line Unix style
- Position from start
• Position from end

1. Select one of the parsing operations from the Operation list in one of the following locations:
   • the Identification Sections or Connectivity Sections for Service Mapping
   • the Step window for Discovery

2. Select one of the keyword, command, or positional types from the Define Parsing list.

3. Click Retrieve.
   The content of the file is displayed in the Output pane.

4. Define the string to extract as a variable from within Debug mode:
   a) Select the string in the Output box.
      All settings are automatically entered in the variable fields.
   b) Provide a unique and meaningful name for the variable and click OK. The variable is added to the Variables table.

5. Define the string to extract as a variable outside of Debug mode:
   a) Specify the value and delimiter for the Keyword, Position from Start, or Position from End.
   b) To add more strings and variables, click the plus icon.

6. To end the discovery process if no results are found, select the If not found check box.

You can use the Command Line Java Style parsing strategy to extract the path of the installation directory of the WebSphere Server.
Parse text using a regular expression

In discovery patterns, you can parse text into variables using **Regular expressions** as the parsing strategy.

- Familiarize yourself with the Regex Java syntax:
  
  [https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html](https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html)

- Navigate to the relevant pattern step:
  
  1. On the pattern form, select the relevant identification section for Discovery.
     
     Alternatively, select the relevant identification or connection section for Service Mapping.
2. Select the relevant pattern step or click to add a step.

Basic knowledge of programming is desirable.
Role required: pd_admin

1. Select one of the parsing operations from the Operation list.
2. Click Retrieve or Retrieve File Content to display the actual file in the Output pane.

3. Select the Regular expression from the Define Parsing list.
4. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular expression</td>
<td>Specify the regular expression. You cannot select text in the text box using this parsing strategy. You can only specify a single expression. To extract multiple values, define several expressions enclosing each expression into parentheses. Matching of variables to parentheses occurs according to the order of the parentheses sets. (The first variable is matched to the first set of parentheses, and so on).</td>
</tr>
</tbody>
</table>
5. To end the discovery process if no results are found, select the **If not found** check box.

---

### Parse text using delimited text

In discovery patterns, you can parse text using the **Delimited text** parsing strategy.

Basic knowledge of programming is desirable.

Role required: `pd_admin`

You can use this parsing strategy for extracting data from any text file. Define which segment of this text file to extract by entering the symbol or word which serves as a boundary and defining the position.

1. Select **Delimited text** from the **Define Parsing** list in one of the following locations:
   - the **Identification Sections** or **Connectivity Sections** for Service Mapping
   - the Step window for Discovery

2. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Include lines with</td>
<td>Specify the constraints used to determine which lines are included in the text selection.</td>
</tr>
<tr>
<td>Exclude lines with</td>
<td>Specify the constraints used to determine which lines are excluded in the text selection.</td>
</tr>
<tr>
<td>Line separator</td>
<td>Specify the non-default character used as a line separator in the text. Default separators are either NEWLINE or CARRIAGE RETURN, depending on the operating system.</td>
</tr>
</tbody>
</table>
3. To define parse the file and extract the string as a variable from within Debug mode:
   a) Select the string to be parsed in the text box. The name box appears.
   b) Provide a unique and meaningful name for the variable and press Enter. The variable is added to the Variables table. The Line Separator, Delimiters, and Positions attributes for the variable are filled in automatically.

4. To define parse the file and extract the string as a variable outside Debug mode:
   a) In the Variables pane, click Add Variable. The new variable with the default name _Name_ shows in the list of variables.
   b) Click the new variable and enter a unique and meaningful name.
   c) If necessary, enter a string in the Line Separator field. This string is used to identify where lines in the file end. For example, common line separators are "/n" or "/r".
   d) Click the Edit icon next to Delimiters, enter the symbol or word to use as a delimiter and click the Plus icon.
e) Specify the position. For multiple positions, use commas as separators.

5. To end the discovery process if no results are found, select the If not found check box.

You can use the Delimited Text parsing strategy for the Parse Variable operation to extract the Queue Manager name of an IBM WebSphere Message Broker.
Customize parsing strategies

You can modify parsing strategies that come as part of the base system or add new ones. Parsing strategies are part of discovery patterns, which Service Mapping and Discovery use to discover and map configuration items (CIs).

Practical knowledge of Java scripting is required.

Role required: pd_admin

Parsing serves to analyze syntax of the source file. You extract values from parsed files, which allows you to convert these values into variables. There are standard parsing strategies, which are part of the base system. If you cannot extract data using the standard parsing strategies, you can create your own parsing strategy. Custom parsing strategies are JavaScript files.

The JavaScript for parsing strategy must comply to the following guidelines:

- You may use the `content` variable to refer to the raw data resulting from executing the operation.
- Use `rtn` to indicate the string, which is the result of your custom parsing.
- The script output must be in well-formed XML syntax.

For example, the output must contain correct tags and markup:

```
<root>
  <OS>Windows</OS>
  <version>10</version>
```
• If necessary, you may use third-party JavaScripts to convert the content into XML.

For example, to convert the content from JSON into XML, use the X2JS JavaScript provided by open-source software.

```javascript
var xtojs = new X2JS();
var result = xtojs.js2xml(jsonObj);
var finalResult = "<root>" + result + "</root>";
```

1. Navigate to Pattern Designer > Custom Parsing Strategies.
2. To modify the JSON custom parsing strategy, select JSON file.
   Alternatively, click New to create a new JSON file.
3. For the new custom parsing strategy, enter a name describing the new strategy in the Name field.
   For example, if the purpose of this strategy is to extract information using a certain protocol, use this protocol as the name.
4. Enter or modify the parsing strategy purpose in the Description field.
5. Write the JavaScript in the Script pane to define the business logic of parsing.
6. Click Update or Submit.

Use this custom parsing strategy for defining operations inside pattern steps.

Define the connection section

Define a set of discovery steps to discover outbound connections of a configuration item (CI). This operation is relevant only for patterns of the application type used by Service Mapping.

Create a pattern or select a pattern that you want to modify as described in Create or modify patterns.

Basic knowledge of programming is desirable.

Role required: pd_admin

A connection section identifies a type of outgoing connection. CIs can have multiple outgoing connections. Configure a separate connection section for each type of outgoing connection. For example, a .NET application CI can have outgoing connections of several types: ADO.NET, XML Web Services, or .NET. So, you must add connection sections for these three types to the .NET Application pattern.

1. Navigate to Pattern Designer > Discovery Patterns and open the required pattern from the pattern list.
   The pattern must be of the Application type.
2. In the Connectivity Section, click New.
   Note: The Connectivity Section is grayed out if the pattern cannot be edited for Service Mapping.
3. Enter the name, and click Save.
4. On the pattern form, click Save to save the pattern.
5. Click the created connection section.
   The pattern designer opens showing the connection step.
6. Select Create Connection from the Operation list.
7. Fill in the fields, as appropriate.
### Field | Description
--- | ---
Select Connection Type | - **Application Flow**: Used between two applications (can be of the same type).
- **Cluster**: Used for connections to CIs of the cluster type. Specify a cluster name.
- **Inclusion**: Used for connections to an object that is included in the current object. For example, a connection from J2EE to EAR, and a connection from IIS to a website.
- **Storage Flow**: Used for connections between configuration items (CIs) of host type and devices of storage type.

Select Entry Point | Select the entry point type from the list. For more information, see [Entry point attributes](#).

Enter Connection Attributes | Define attributes by either entering actual values or variables.

Select Target CI Type | Select the target CI type from the list.

Is hidden | Select the check box if the connection should be hidden (that is, not shown in the user interface), but is used for continuing the discovery flow.

Is traffic based | Select the check box to use the traffic-based discovery method for this connection.

---

8. If necessary, create more connectivity sections on the pattern form.

<table>
<thead>
<tr>
<th>This operation is used in</th>
<th>This item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td>Applications &gt; Business Integration Software</td>
</tr>
<tr>
<td>CI Type</td>
<td>IBM WebSphere MQ Queue</td>
</tr>
<tr>
<td>Pattern</td>
<td>WMQ Queue Unix Pattern</td>
</tr>
<tr>
<td>Section</td>
<td>Alias queues connectivity</td>
</tr>
<tr>
<td>Step number and Name</td>
<td>6. Create outgoing connection to alias queues</td>
</tr>
</tbody>
</table>

To create a connection from Microsoft Exchange CAS to Exchange Mailboxes, define the **Create Connection** operation as follows:
Example of creating an application pattern

Follow this example to see a step-by-step process of creating and defining the identification section for a new application pattern.

Basic knowledge of programming is desirable.

Role required: pd_admin

This example shows how to create a mapping pattern using the debug mode.

The pattern is for Apache Web Server on Unix.

1. Navigate to Pattern Designer > Discovery Patterns.
2. Click New.
3. Define the basic pattern attributes as follows:
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern type</td>
<td>Select Application.</td>
</tr>
<tr>
<td>Name</td>
<td>Enter Apache Web Server on Unix Pattern.</td>
</tr>
<tr>
<td>CI Type</td>
<td>Select Apache Web Server from the list.</td>
</tr>
<tr>
<td>Operating system (Application patterns only)</td>
<td>Clear the All check box and select the following check boxes from the list:</td>
</tr>
<tr>
<td></td>
<td>• AIX Server</td>
</tr>
<tr>
<td></td>
<td>• HPUX Server</td>
</tr>
<tr>
<td></td>
<td>• Linux Server</td>
</tr>
<tr>
<td></td>
<td>• Solaris Server</td>
</tr>
<tr>
<td></td>
<td>• UNIX Server</td>
</tr>
<tr>
<td>Run Order (Application patterns only)</td>
<td>Leave the default None setting.</td>
</tr>
<tr>
<td>Description</td>
<td>This pattern discovers Apache Web Servers on Unix versions up to 2.4.</td>
</tr>
</tbody>
</table>

4. Click Save.
5. Create the identification section and define its basic properties:
   a) Under Identification Section, click New.
   b) Configure the following parameters:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter Identification for HTTP(S) entry point type(s).</td>
</tr>
<tr>
<td>Entry Point Types (Application patterns only)</td>
<td>Select the following check boxes from the list:</td>
</tr>
</tbody>
</table>
**Find Process Strategy** (Application patterns only) | Select Listening Port.
---|---
**Order** | Enter 1.

6. Click the newly created identification section: **Identification for HTTP(S) entry point type(s)**. The new identification section opens on the separate tab.

7. Activate the debug mode:
   a) In the Pattern Designer, click **Debug Mode**.
      The Debug Identification Section window is displayed.
   b) Fill in the required details for the entry point type:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debug Type</td>
<td>Select <strong>Top down</strong> for performing top-down discovery with Service Mapping or</td>
</tr>
<tr>
<td>Type</td>
<td>Select HTTP(S) for the entry point type from the list.</td>
</tr>
<tr>
<td>URL</td>
<td>Enter <a href="http://10.196.39.244:6080/ITO">http://10.196.39.244:6080/ITO</a>.</td>
</tr>
</tbody>
</table>

   c) Click **Connect**.
   The debug mode is activated and the green dot appears on the debug button:

   ![Debug Mode - Top down](image)

   Notice that the following variables are populated with values once the debug mode is active:
   - computer_system # the Apache host information
   - entry_point # identified by the URL in this case
   - process # the Apache process information

8. Check that the process name on the CI is Apache Web Server:
   a) Rename the first step of the identification section to **Check process name to match Apache**.
   b) Select **Match** from the **Operation** list.
   c) Enter `$process.executable` in the first condition field.
d) Select Contains from the conditional operator list.

e) Enter "httpd" in the second condition field.

f) Click the plus icon to add another condition.

g) Enter $process.executable in the first condition field.

h) Select Contains from the conditional operator list.

i) Enter "apache" in the second field.

j) Define that this match operation must match one of these conditions: Select Any from the Meet list.

k) Click Test and verify that you get the following message: No changes were made during this test.

9. Populate the label attribute for your CI:

   a) In the Steps tree, click to add a step blow the first step.
   b) Rename the new step to sets the display label.
   c) Select Set Parameter Value from the Operation list.
   d) Enter "Apache" in the Value field.
   e) Enter $name in the Name field.
   f) Click Test and verify that the following message appears:
g) Click **Close**.

10. Populate the home directory attribute:
   a) In the **Steps** tree, add a step and rename it to **Get home dir**.
   b) Select **Parse variable** from the **Operation** list.
   This operation extracts the value after the `-d` in the content box.
   c) Expand the **process** variable in the **Temporary Variables** pane.
   d) Drag the `commandLine` variable from the Temporary Variables pane into the variable field under operation.
Note: For more information on using the drag-and-drop feature, see Enter values and variables in patterns.

e) Select Command line Unix style from the Define Parsing list.

f) In the Variables pane, add the new install_directory variable.

g) Click Test.

11. Obtain the home directory attribute from the HTTP daemon.
If the previous step populated the home directory attribute, skip this step. In this example, it is necessary to perform it.

a) In the Steps tree, add a new step and rename it to Condition – check that home dir was set if not extract it from httpd -V.

b) Select Parse command output from the Operation list.
This operation extracts the value after the -d in the content box.

c) Click Precondition.

d) Enter $install_directory in the condition value field.

e) Select Is Empty from the conditional operator list.

f) Select True from the If condition Is list.

g) Enter $process.executablePath + " -V " in the Set Command Details field.

h) Click Run Command.

i) Select Delimited text from the Define Parsing list.

j) Enter HTTPD_ROOT in the Include lines field.
k) Click the **Edit** button next to **Delimiters**.

![Edit Delimiters window](image)

l) Add the two delimiters: equals (=) and quotes (").

m) Click **OK**.

n) Enter 2 in the **Positions** field.

o) Click **Test**.

   The Debug Results window shows the home directory attribute populated with a value.

p) Click **OK**.

12. Populate the CI configuration file attribute:
   a) In the **Steps** tree, add a new step and rename it to **Get config file**
   b) Select **Parse variable** from the **Operation** list.
   c) Expand the **process** variable in the **Temporary Variables** pane.
   d) Drag the commandLine variable from the **Temporary Variables** pane into the **Enter Variable** field.

   The value is populated: `/usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf`.

   e) Select **Command Line Unix Style** from **Define Parsing** list.
f) Click Retrieve.
g) Enter -d in the Keyword field.
h) Select the value in the Output pane and create a new variable named install_directory.
i) Click Test.

13. Extract the CI configuration file attribute from the HTTP daemon:
If the previous step populated the configuration file attribute, skip this step. In this example, it is necessary to perform it.
a) In the Steps tree, add a new step and rename it to Condition – check that conf_file was set if not extract it from httpd -V.
b) Select Parse command output from the Operation list.
c) Click Precondition.
d) Enter $config_file in the condition value field.
e) Select Is Empty from the conditional operator list.
f) Enter $process.executablePath" -V " in the Set Command Details field.

g) Click Run Command.

h) Select Delimited text from the Define Parsing list.

i) Enter SERVER_CONFIG_FILE in the Include lines field.

j) Click the Edit button next to Delimiters.

k) Add the two delimiters: equals (=) and quotes (").

l) Click OK.

m) Enter 2 in the Positions field.

n) If the new conf_file variable is not added automatically, create it in the Variables pane.

o) Click Test.
The Debug Results window shows the configuration file attribute populated with a value.

p) Click OK.

14. If the configuration file attribute is still not populated, perform this step:
   a) In the Steps tree, add a new step and rename it to default location of conf file.
   b) Select Set Parameter Value from the Operation list.
   c) Click Precondition.
   d) Enter $conf_file in the condition value field.
   e) Select Is Empty from the conditional operator list.
   f) Select True from the If condition is list.
   g) Enter $home_dir+"/conf/httpd.conf" in the Value field.
   h) Enter $conf_file in the Name field.
   i) Click Test and verify that the configuration file attribute is populated.

15. Concatenate the home directory and configuration file values:
   a) In the Steps tree, add a new step and rename it to check if the SERVER_CONFIG_FILE is relative or not.
   b) Select Set Parameter Value from the Operation list.
   c) Click Precondition.
   d) Enter $conf_file in the condition value field.
   e) Select Starts With from the conditional operator list.
   f) Enter "/" in the string value.
   g) Select False from the If condition is list.
   h) Enter $home_dir+"/$conf_file in the Value field.
   i) Enter $conf_file in the Name field.
   j) Click Test and verify that the configuration file attribute is populated.

16. Populate the version attribute:
   a) In the Steps tree, add a new step and rename it to get version from version.signature (IBM HTTPSERVER).
   b) Select Parse file from the Operation list.
   c) Enter the concatenated $install_directory variable and "/version.signature" string ("$home_dir+/version.signature") in the Select File field.
   d) Click Retrieve File Content.
   e) Create the Version variable in the Variables pane.
   f) Click Test and verify that the version attribute is populated.
   In this example, the version is not extracted at this stage.

17. Extract the version attribute from the HTTP daemon:
   a) In the Steps tree, add a step and rename it to Condition - check that version was set if not extract it from httpd -v.
   b) Select Parse command output from the Operation list.
   c) Click Precondition.
d) Enter $version in the condition value field.
e) Select Is Empty from the conditional operator list.
f) Click the plus icon (+) to add another condition.
g) Enter $version in the condition value field.
h) Select Contains from the conditional operator list.
i) Enter "directory" in the string value.
j) Select Any from the Meet list.
k) Select true from the If precondition is list.

l) Enter $process.executablePath+" -V | grep 'Server version' | cut -d '/' - f 2 | cut -d ' ' -f 1" in the Set Command Details field.
m) Click Run Command and verify that the version attribute appears in the Output pane.

n) Select Position From End from the Define Parsing list.
o) Enter 1 in the Positions field in the Variables pane.
p) Click **Test** and verify that the configuration file attribute is populated.

Note:
The version number appears only in the **Temporary Variables** pane, not in the **CI Attributes** pane.
Notice that at this stage you have successfully identified the Apache Web Server and populated its various attributes except the version attribute that is left blank on purpose.

18. If the version is still not populated, extract it from the IHS.product file:
   a) In the Steps tree, add a step and rename it to set version if still empty.
   b) Select Parse File from the Operation list.
   c) Click Precondition.
   d) Enter $version in the condition value field.
   e) Select Is Empty from the conditional operator list.
   f) Click the plus icon (+) to add another condition.
   g) Enter $version in the condition value field.
   h) Select Contains from the conditional operator list.
   i) Enter "directory" in the string value.
   j) Select Any from the Meet list.
   k) Select true from the If precondition is list.
   l) Enter $install_directory + "/properties/version/IHS.product" in the Select File field.
   m) Click Retrieve File Content.
      The content of the file is displayed in the Output pane.
   n) Select XML File from the Define Parsing list.
   o) Mark the version value in the Output pane, and then enter version in the variable pop-up.
19. Find additional attributes by reusing a shared step library.
   a) In the Steps tree, add a step, and rename the new step to reference to enrich library.
   b) 
   c) Select Library reference from the Operation list.
   d) Select Apache Enrich Attributes from the Library list.
      This operation inserts a sequence of preconfigured substeps into the step tree. For information on how to create shared step libraries, see Reuse a shared step library.
20. Populate process-related attributes:
   a) In the **Steps** tree, add a step and rename it to **get processes**.
   b) Select **Get process** from the **Operation** list.
   c) Enter "httpd" in the **Command Line** field.
   d) Enter in the $procs in the **Specify Target Variable** field.
   e) Click **Test** and verify that the attributes are displayed:
21. Discover the process IDs:
   a) In the **Steps** tree, add a step and rename it to **set process_ids**.
   b) Select **Parse variable** from the **Operation** list.
   c) Enter `$procs[*].pid` in the **Enter Variable** field.
   d) Select **Delimited text** from the **Define Parsing** list.
   e) Click **Retrieve**.
      The content is displayed in the **Output** pane.
   f) Create the **process_ids** variable in the **Variables** pane.
   g) Enter 1 in the **Positions** field.

f) Click **Close** when done.
h) Click **Test**.

i) Verify that all the necessary attributes are populated:
22. Click **Save**.

23. Verify that Discovery and Service Mapping can use the identification section you defined:
   a) Run the horizontal discovery of an Apache Web Server using this pattern.
   b) Navigate to the table of Apache Web Servers and check that there is an entry for this CI.
   c) Run the top-down discovery of the same CI.
   d) Check the same Apache Web Servers table.
   e) Verify that the record is not duplicated.
It means that the results of both horizontal and top-down discovery are written in the CMDB under the same record. The identification section of the pattern is correct.

Discover related items together with the main CI

Add related items to the patterns to perform horizontal discovery of configuration items with all their related items: CIs or non-CIs, like ports or serial numbers.

1. (Optional) Check if the CI is dependent: The system uses the identification rules to identify the CI independently of other CIs. Refer to Create or edit a CI identification rule.

2. (Optional) To understand the model better, check dependent relationship rules which define the dependency structure of the CI types and the relationship types in these service definitions. For example, you can discover a Tomcat server with Tomcat WARs belonging to it. For more information, refer to CMDB dependent relationship rules.
Example of the Containment rule for the Tomcat WAR CI type

3. To add a non-CI to the pattern, add it as a related item for the CI type to which the main CI belongs. For more information on adding related items for the CI type, see Configure references to other CIs or non-CIs as related items for the CI type.

Role required: pd_admin

Related items, both CIs and non-CIs, serve as a reference and provide additional information about the CI. For example, you can define a serial number, a port and an network APR table as
related items for a hardware server. The system does not use these items for identification. After you configure related items for the CI type, you can use horizontal discovery for CIs of this type together with their related items.

**Note:** Non-CIs are items that do not extend the Configuration Item (cmdb_ci) table. Examples of non-CIs are ports, serial numbers, or Network ARP tables.

Pattern Designer displays attributes of related CI types, which you added to the pattern in the CI attributes pane. You can view and use these attributes while creating or modifying the main CI pattern.

1. Navigate to Pattern Designer > Discovery Patterns.
2. Select the relevant pattern.
3. On the pattern form, select the relevant identification section.
4. Make sure that you are not in Debug mode.
5. On the CI Attributes pane, click the Manage related CIs icon.

![](image)

6. Select the relevant related CI type.
   You can use the search box to locate a specific CI type.
7. Click the Plus icon.
   The new related CI type is added to the list.
8. Optional: Add other related CI types.
9. Click OK.
   New tabs for related CI types you added appear under the tab for the main CI:
Continue editing the pattern by adding a new step and defining its operation. To view or use attributes of a related CI type, click the relevant tab in the CI Attributes pane.

Enhance patterns without changing their identification sections

Enable patterns to search for additional attributes and modify pattern discovery logic defined in identification sections by using extension sections.

If your ServiceNow instance uses domain separation and you have access to the global domain, select the domain to which the application service belongs from the domain picker. The selected domain must be a domain without any child domains.

Basic knowledge of programming is desirable.

Role required: pd_admin

Patterns can be of infrastructure or application type. Infrastructure patterns are used only by Discovery for creating lists of devices. Application patterns serve both Service Mapping and Discovery that use the same application patterns for their purposes. Each pattern including its identification and connection sections is an individual file. There are also reusable sequences of discovery steps referred to as shared libraries. Each shared library is an individual file, but it is always part of the infrastructure or application pattern and is not used as a standalone pattern. Records for both patterns and shared libraries reside in the Discovery Patterns (sa_pattern) table.
To modify the discovery logic in identification sections without changing the identification sections themselves, you can add extension sections to the pattern. Every extension section consists of a shared library, either new or existing.

When you create or modify a shared library in the extension section, you save changes only to the shared library, not to the pattern containing this extension section.

⚠️ **Warning:** The changes to the shared library affect all patterns using this shared library.

The MID Server runs extension sections after identification sections and only if at least one identification section completes successfully.

If there are more than one extension sections associated with this pattern, define the order in which the MID Server runs these extension sections.

You can deactivate shared libraries, so that patterns using them, skip steps in deactivated shared libraries. For more information, see *Discovery Configuration Console*.

1. Navigate to **Pattern Designer > Discovery Patterns** and open the required pattern from the pattern list.
   The pattern can be of the Infrastructure or Application type.
2. To define discovery steps for the new extension section.
   a) Click **New** under **Extension section**.
   b) Enter the name of this extension section.
   c) Click **Done**.
      The pattern form displays the new section under **Extension section**.
   d) Click the name of the new extension section.
      The Pattern Designer opens showing the Steps tree on the left.
      If no discovery steps have been identified for this pattern, the **Untitled Step** appears in the Steps tree in the left pane of the window.
   e) **Define discovery steps**.
   f) Click **Save** to save the extension section and make Service Mapping and Discovery use it.
You do not publish extension sections

3. To use an existing shared library for the new extension section:
   a) Click **Add** under **Extension section**.
   b) Select the shared library from the name list.

4. If there are more than one extension sections, define the order in which the MID Server runs the extension section.
   a) Select the check box next to the extension section.
   b) Click **Edit** under **Extension section**.
   c) Enter a number that determines the order in the **Order** field.
   If there is only one extension section, its order is 1 by default.
   The section with the lowest order number is used first.
   d) Click **Done**.

**Click Debug** to run the newly created extension section. Pattern Designer runs extension sections after running identification sections.

**Fine-tune patterns using traffic-based discovery**

As an alternative to customizing the pattern from the Pattern Designer side, improve existing patterns so that Service Mapping can use them to find configuration item (CI) connections previously found using traffic-based discovery.

Basic knowledge of programming is desirable.

Role required: pd_admin

Not only can Service Mapping discover CIs using patterns, it can also discover them by following traffic connections between CIs. This method is referred to as traffic-based discovery.

Using traffic-based discovery, Service Mapping may discover an important CI connection. In this case, you can make sure that this connection is always found for all CIs belonging to the same CI type. You do that by modifying the pattern used to discover CIs of that CI type. A pattern is a sequence of steps whose purpose is to detect attributes of a CI and its outbound connections.

You can search CI configuration files to locate data discovered using traffic-based discovery, for example IP address and port. If there are configuration files containing this data, you create a pattern step to parse this file. Next time Service Mapping discovers this CI using the modified pattern, it extracts the information about this connection directly from the configuration file.

1. Alternatively, if Service Mapping is deployed, navigate to **Service Mapping > Services > Application Services**.
2. Click **View Map** next to the relevant application service.
3. Right-click the relevant CI.
4. Select **Show traffic based connections**.
   The Traffic Based Connections List opens displaying the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>The IP address of the application connected to the selected CI.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Port</td>
<td>The port on the selected CI that is used to communicate to the other application.</td>
</tr>
<tr>
<td>Process</td>
<td>The ID of the process in the selected CI.</td>
</tr>
<tr>
<td>Already on map</td>
<td>• Yes — if this connection shows on the map.</td>
</tr>
<tr>
<td></td>
<td>• No — if this connection is not part of the application service and not on the map.</td>
</tr>
<tr>
<td>System decision</td>
<td>The setting defines if Service Mapping keeps the discovered traffic-based connection or removes it. The value comes from the algorithm that Service Mapping uses.</td>
</tr>
<tr>
<td>User decision</td>
<td>(Optional) The setting overrides the System decision setting, which defines if Service Mapping keeps the discovered traffic-based connection or removes it. For example, if the System decision setting for a connection is Keep, and you want to remove this connection, select Remove.</td>
</tr>
</tbody>
</table>

**Note:** There may be a case when you can see traffic-based connections on the map, but the Traffic Based Connections list does not display them. It happens for the connections that have been removed from the TCP Connection (cmdb_tcp) table less than three days.

5. Double-click the connection you want to use to create a pattern step.

6. If there are multiple patterns used to discover this CI, select the pattern for which you want to create the step, and click **OK**.

Pattern Designer creates a temporary connection section for this pattern with a new step.
The Debug Connection Section window is displayed with some fields automatically populated.

7. Click **Connect**.

The Debug mode is activated.

The Search Assistant window opens.

8. Search CI configuration files for data received using traffic-based discovery:
   a) Under **Free Text**, review data automatically filled in from the Traffic Based Connections List.
The File Patterns field displays file types in which Pattern Designer performs the search.

b) Click the **Search Files** tab and review files. If necessary, you can remove irrelevant files from the search by clicking the Minus icon.

c) Click **Search**.

After search completes, the **Search Result** tab opens with the list of files in which any of the search criteria was found.

9. Create a pattern step for parsing the configuration file containing information about the connection:

a) On the Search Result tab, click one of the files and review connection data found in this file.

b) Wait for the file content to load and select the relevant entry under the file name:
c) To use this data to create the pattern step, click **Create parse file step**.

Pattern Designer shows the new step with the Parse File operation. It automatically assigns this step the name in the following format: Parsing <file name>.

Pattern Designer automatically sets the file name, file path, and parsing strategy for the step. It also creates variables that contain data you searched for. If there is a variable, which contains the file path, Pattern Designer enters this variable for the path.
d) Optional: If necessary, you can modify parsing strategy.
e) Rename the section to provide a meaningful name, for example Discover HA Proxy client connection.

Or

Copy and paste the step into the relevant connection section.

10. Complete modifying the pattern as described in **Finalize a pattern**.

**Finalize a pattern**

After you finish defining your pattern, make it ready for use by Service Mapping and Discovery.

Role required: pd_admin

There are several ways to finalize a pattern after creating or modifying it:
• Saving — Save all the information you defined for the pattern creating a draft. You can continue working on the pattern draft at any time. Service Mapping and Discovery do not use the pattern draft for discovery and mapping.

• Discarding changes — If you realized that the changes you made to a pattern are incorrect, you can discard them.

• Checking — Actually run the pattern on the relevant configuration item (CI) to check that the operations that you defined for it are correct. Check your pattern before activating it. This option is available only for application patterns used by Service Mapping.

• Publishing — Make Service Mapping and Discovery use this pattern for discovery and mapping. A new version of this pattern is created. The state of the pattern version used previously changes to History.

1. On the pattern definition form, complete your pattern definition by performing one or more of the following actions:

<table>
<thead>
<tr>
<th>To accomplish this</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save the pattern</td>
<td>Click <img src="Save.png" alt="Save" />.</td>
</tr>
<tr>
<td>Discard unwanted changes</td>
<td>Click <img src="Discard.png" alt="Discard" />.</td>
</tr>
</tbody>
</table>
| Check the pattern (only for application patterns used by Service Mapping) | 1. Click ![Check Pattern](Check Pattern.png).  
2. Enter parameters for connecting to this CI.  
3. Click ![Connect](Connect.png).  
4. Wait for Pattern Designer to run the pattern and check that there are no errors.  
5. Troubleshoot any errors. If there are no errors, you can publish or save your pattern. |

| Publish the pattern | Click ![Publish](Publish.png). |

2. Close the Pattern Designer tab.

**Copy patterns from one instance to another**

To copy patterns from one instance to another, create an update set containing new or modified patterns with their related items in your development instance. Then, import the update set into your production instance.

Role required: pd_admin
Typically, you maintain two ServiceNow instances in your organization: for production and for development. Create or modify patterns, test them, and verify results in the development instance. When you are satisfied with the discovery results, export relevant patterns from the development instance to create an update set.

Update sets is a ServiceNow feature for transferring customizations from one instance to another. Each update set is stored in the Update Set (sys_update_set) table. Specific customizations associated with the update set are the entries in the Customer Update (sys_update_xml) table, and they appear as a related list on the update set record. The method of creating update sets for patterns is different from creating update sets for other types of customizations.

**Warning:** Always follow this procedure to create update sets for patterns. Do not use the standard platform procedure for creating update sets.

Depending on the extent of the changes, you can use the following pattern export methods:

### Export patterns

This option creates an update set containing the modified patterns and the related items necessary for the patterns to work. The related items include:

- Extension sections
- Tracked files
- Related CI types
- Classification records
- Associated scripts

**Note:** Update sets do not include changes in script includes used in patterns.

### Export patterns with CMDB items

This option creates an update set that contains modified patterns with the related items from the previous option and the CMDB items associated with the modified patterns. The CMDB items include:

- Main CI type
- Entry point types
- Lookup tables
- Identifiers
- Hosting and reconciliation rules

The update set is an XML file with the following name convention:

```
sys_remote_update_set<random alphanumeric string>.xml.
```

1. In the development instance, perform these steps to export patterns:
   a) Navigate to Pattern Designer > Discovery Patterns.
   b) Select application or infrastructure patterns that you want to export.

**Note:** You cannot export shared libraries.

   c) At the bottom of the window, click the Actions on selected rows button.

**Note:** You cannot use more than 20 patterns at a time for the Export patterns with CMDB items method.
d) Click **Export patterns** or **Export patterns with CMDB items**.

e) When the system finishes exporting patterns, it creates the update set.

f) To view the list of items included in the update set, click **View Logs** in the success message.

g) Click **Close**.

The update set file in XML format is automatically downloaded on the computer.

h) Save the update set file locally for future use.

2. In the production instance, perform these steps to import the customized patterns:

   a) Elevate privileges to the security_admin role.

   b) Navigate to **System Update Sets > Retrieved Update Sets**.

   c) Click the link **Import Update Set from XML**.

   d) Click **Choose File** and select an XML file.

   e) Click **Upload**.

   The customization is now available as a retrieved update set with state Loaded.

3. **Preview a remote update set**.

   The system scans and validates the update set and displays its contents on the **Customer Updates** tab.

4. **Commit the update set**.

5. Click **Close** in the success message.

   The production instance contains all new or modified patterns with related items from the update set.

6. Upload the updated set of patterns and related items onto the MID Server:

   a) Navigate to **Discovery > MID Servers**.

   b) Alternatively, navigate to **Service Mapping > MID Servers**.

   c) Click **Pattern Sync to Mid**.

7. If the discovery fails or the result is not as expected, restart the MID Server to make sure that the MID Server runs the latest set of patterns.

**Choose the pattern version**

Every time you modify and save a pattern, you create a version of this pattern. Choose which pattern version Service Mapping and Discovery use for discovery.

Role required: pd_admin

By default, the latest version is used for discovery, but you can choose any other version to use.

If necessary, you can **compare pattern versions**.

1. Navigate to **Pattern Designer > Discovery Patterns**.

2. Click the required pattern.

   The form for this pattern opens.

3. Click the **Pattern** tab.

4. Scroll down to the **Versions** section.

   The state of the active pattern version that Service Mapping and Discovery use is **Current**.

5. Click the version you want to use.
6. Click **Revert to this version** under **Related Links**.

This pattern version becomes the active version that Service Mapping and Discovery use. Its state changes to Current.

**Compare pattern versions**

If you have multiple versions of the same pattern, you can compare them to decide which pattern version to use for discovery.

Role required: pd_admin
Service Mapping and Discovery use patterns in their discovery process. Every time you modify and save a pattern, you create a version of this pattern.

1. Navigate to **Pattern Designer > Discovery Patterns**.
2. Click the required pattern.
   The form for this pattern opens.
3. Click the **Pattern** tab.
4. Scroll down to the **Versions** section.
   The state of the active pattern version that Service Mapping and Discovery use is **Current**.
5. Click the previous pattern version, which you want to compare to the current version.
6. Click **Compare to Current** under **Related Links**.
7. In the Compare to Current window, click the **Pattern text** pane either under **Selected Version** or **Current Version**.

![Compare to Current Window](image)

The text comparison view opens showing the text of the selected pattern version and the current pattern version side by side. The step containing a difference appears highlighted. Highlight colors indicate the type of change:

- Updated step — purple
- New line or characters — green
- Deleted line or characters — red

8. Review the differences in the pattern versions.
9. If necessary, click **Revert chunk** to replace the step in the current version with the step from the previous version.
10. Click OK when done.

11. If you copied any steps from the previous version and want to save the change, click Save Merge.

**ITOM Health**

ServiceNow® ITOM Health product includes the ServiceNow® Event Management feature and the ServiceNow® Operational Intelligence feature, which together help you to track and maintain the health of services in your organization.

**Features of ITOM Health**

**Event Management**

Event Management gathers alerts from infrastructure events captured by third-party monitoring tools. Event Management uses IT-related information gathered by Discovery to map alerts to configuration items. Based on the collected information, Event Management then provides dashboards showing a consolidated view of all service-impact events.

**Operational Intelligence**

Use Operational Intelligence to proactively analyze your IT infrastructure to spot issues and prevent service outages. Using advanced machine learning to analyze information about your IT infrastructure, the application automatically determines dynamic thresholds and identifies anomalies that may indicate potential service outages.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Using guided setup to implement IT Operations Management applications

IT Operations Management Guided Setup provides a sequence of tasks that help you configure IT Operations Management applications on your ServiceNow instance. To open IT Operations Management guided setup, navigate to Guided Setup > ITOM Guided Setup. For more information about using the guided setup interface, see .

Event Management

The ServiceNow® Event Management helps you to identify health issues across the datacenter on a single management console. It provides alert aggregation and root cause analysis (RCA) for discovered services, application services, and automated alert groups. Event Management is available as a separate subscription from the rest of the ServiceNow platform.

Understanding Event Management

Monitor the health of business services and infrastructure using a single management console and respond appropriately to any issues that come up. Event Management provides intelligent event and alert analysis to ensure continuity of your business service performance. Event Management receives and processes events via the MID Server.

What Event Management can manage

Event Management can manage:

Discovered business services

A Business service is a definition of interrelated CIs from the CMDB. The discovered service, from Service Mapping, includes a business service map with: mapping relationships, an impact tree showing outage severity, active alerts, related alerts, and CI properties. Business
service information is discovered by Service Mapping. The mapping information appears on dashboards, the Alerts list, and the Events list.

**Application service**

An application service is a business service created by selecting CIs to include in the service. Application service information appears on dashboards with drill-down capability to a map view.

**Technical services**

A Technical service is a dynamic grouping of CIs, based on some common criteria. For example, you can create a technical service based on location for all web servers or all databases in Ireland.

**Alert groups**

Alert groups show sets of alerts for ease of maintenance. Alert groups are not the same as automated alert groups.

**Architecture**

As events occur on various systems, the MID Server connector instance sends the events to the instance. Event Management generates alerts, applies alert management rules, and prioritizes alerts for remediation and root cause analysis. View this information on dashboards, the alert list in Alert Intelligence, or from a service map.
Workflow

Event Management receives external events and generates alerts based on event and alert management rules. Events are sent directly to your instance using an email server, script, SNMP trap, or a web service API. The corresponding alerts appear on dashboards for tracking and remediation purposes.

As the computer, software, or service generates events, the MID Server polls the external event tracking tool. The MID Server, which maintains a connection to Event Management, sends the information to your instance for storage, processing, and remediation.

The instance stores events in the Event (em_event) table and attempts to generate alerts based on pre-defined rules and event mappings. Regardless of whether an alert generates, the original event is available for review and remediation. Alerts generate according to the following process flow:

1. Find the best matching event rule for an event. If the source of the event matches the source specified in an existing rule, then a rule is matched. Also, if the event matches the optional rule Filter and the event additional_info value matches the rule Additional Information filter. A rule without any filter is ignored, for example the source filter is missing or the Additional Information filter is missing. If multiple rules are defined for the same type of event, use the rule Order to determine the order of rule application.
• If the rule **Ignore** check box is selected, no alert generates. However, the event is still available for review and remediation.
• If transforms have been defined, apply them. If compose parameters are set, apply the additional content to display to the user in the alert.
• If **Active** in the threshold section is selected, accumulate all events until the threshold is met. Generate a single alert for the events.

2. Search for an event field mapping even if there was no event rule. If an event field mapping is found, apply the mapping information. If the event has no severity after the event transformations, retain the event for reference purposes and do not generate an alert.

3. Search the Alert (em_alert) table for a matching message key. If a matching message key exists, update the alert according to the event information. If a matching message key does not exist, create an alert. If another event has the same matching key, associate the events under a single alert. For root cause analysis purposes, bind the alert to a specific CI.
Event Management and Service Mapping

Event Management uses discovered services from Service Mapping and automated alert groups with root cause analysis (RCA) to expedite alert resolution.

When an event from an external source arrives from the MID Server, script, or web service API (not pictured), Event Management locates CI information for alert generation and CI remediation. CI information is stored in the CMDB from sources such as Service Mapping, Discovery, third-party sources, and manual population. You can use correlated alert group and root cause analysis information to resolve the issue.
Event Management interoperability

Request Event Management

Event Management plugin (com.glideapp.itom.snac) requires a separate subscription and must be activated by ServiceNow personnel. This plugin includes demo data and activates related plugins if they are not already active.
Role required: admin

To purchase a subscription, contact your ServiceNow account manager. The account manager can arrange to have the plugin activated on your organization's production and sub-production instances, generally within a few days.

If you do not have an account manager, decide to delay activation after purchase, or want to evaluate the product on a sub-production instance without charge, follow these steps.

1. From your instance, navigate to System Definition > Plugins.
2. On the All Applications page, click Request Plugin to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>Date and time must be at least two business days from the current time.</td>
</tr>
<tr>
<td>Reason/Comments</td>
<td>Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments.</td>
</tr>
</tbody>
</table>

5. Click Submit.

Enhance Event Management performance

The Event Management Accelerator plugin ensures that Event Management maintains performance at a high level. This plugin is optional.

Role required: admin

The `em_alert_history` and `em_impact_status` tables can grow to be very large, negatively impacting Event Management performance. Installing the Event Management Performance Accelerator plugin ensures that Event Management performs at a high level.

When upgrading from an earlier version with more than 5 million records in either the `em_alert_history` or `em_impact_status` tables, you must activate this plugin manually. When there are less than 5 million records, it runs automatically. You can customize this number by modifying the `evt_mgmt.plugin_activation.table_max_size` property.

To purchase a subscription, contact your ServiceNow account manager. The account manager can arrange to have the plugin activated on your organization's production and sub-production instances, generally within a few days.

If you do not have an account manager, decide to delay activation after purchase, or want to evaluate the product on a sub-production instance without charge, follow these steps.

1. From your instance, navigate to System Definition > Plugins.
2. On the All Applications page, click Request Plugin to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>Date and time must be at least two business days from the current time.</td>
</tr>
<tr>
<td>Reason/Comments</td>
<td>Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments.</td>
</tr>
</tbody>
</table>

5. Click Submit.

Out-of-the-box Event Management Performance Analytics Solutions

Performance Analytics Solutions contain preconfigured dashboards. These dashboards contain actionable data visualizations that help you improve your business processes and practices.

Performance Analytics Solutions

Use the Performance Analytics widgets on the dashboard to visualize data over time, analyze your business processes, and identify areas of improvement. With solutions, you can get value from Performance Analytics for your application with minimal setup.

**Important:** Set up and test Out-of-the-box Performance Analytics Solutions on a sub-production instance before enabling them in production. You can set up and test Performance Analytics on a sub-production instance without a subscription.

**Note:**
- Solutions include some dashboards that are inactive by default. You can activate these dashboards to make them visible to end users according to your business needs.
Out-of-the-box solutions and in-form analytics provide all the configuration records required to analyze default applications. Customize these records for use in your production environment.

To enable the solution for Event Management, an admin can navigate to Performance Analytics > Guided Setup. Click Get Started then scroll to the section for Event Management. The guided setup takes you through the entire setup and configuration process.

Implement Event Management in your instance

You can set up Event Management, including a MID Server, either with or without using guided setup in your instance.

Using guided setup to implement Event Management

Event Management guided setup provides a sequence of tasks that help you configure Event Management on your ServiceNow instance. To open Event Management guided setup, navigate to Guided Setup > ITOM Guided Setup. For more information about using the guided setup interface, see Using guided setup.

Setup to implement Event Management without using guided setup

Follow the specified tasks to setup Event Management. For more information, see Event Management setup.

Event Management setup

After activating Event Management, set it up to receive and process events, and generate and analyze alerts.

Event Management setup without using guided setup

Set up Event Management by completing these tasks in the following order:

1. Navigate to Event Management > Guided Setup and follow the configuration steps.
2. Configure a MID Server to receive and process events via the MID Server.
3. Configure connector definitions and connector instances to receive external events.
4. Configure event field mappings and alert binding to manage alert generation.
5. Alert management rules for resolving alerts, perform alert remediation, and CI remediation for alert management.
6. Get a top-down discovery to receive CI relationships for software and hardware.
7. Configure impact calculation for services, to establish priority for alert resolution.
8. Configure alert groups to consolidate related alerts.
9. Enable root cause analysis (RCA) for business services and for application services:
   a. Configure a dedicated RCA MID Server as described in Legacy: Configure a MID Server for RCA for business services.
   b. Set the sa_analytics.rca_enabled property to true.
c. To enable RCA for discovered business services, ensure that Service Mapping is enabled as described in Request Service Mapping. This step is not required for RCA for application services.

d. Request Service Mapping

e. Legacy: Configure a MID Server for RCA for business services

10. Configure any other general tasks that appear in this section as appropriate.

Note: Event Management does not support creating incidents on remote instances.

Note: Event Management does not support mobile devices.

Event Management setup using guided setup

Event Management guided setup provides a sequence of tasks that help you configure Event Management on your ServiceNow instance. To open Event Management guided setup, navigate to Guided Setup > ITOM Guided Setup. For more information about using the guided setup interface, see Using guided setup.

Event Management during a platform upgrade

During a platform upgrade Event Management jobs whose Upgrade safe flag is marked as true remain running.

During an upgrade of the platform, the Event Management connectors are working and continue to retrieve events. The events are being processed, transformed to an alert using event rules and if there is a suitable alert management rule, a task is created. However, the impact calculation is not supported during the platform upgrade.

Note: Event Management jobs that started running before the platform upgrade commenced continue to run during the upgrade.

The following Event Management jobs remain running as their Upgrade safe flag is marked as true:

- Event Management - Connector execution
- Event Management - Update stuck connect
- Event Management - Alert Priority Queue
- Event Management - close flapping alerts
- Event Management - close threshold alert
- Event Management - Evaluate Alert Management
- Event Management - Maintenance Calculator
- Event Management - process events

During the platform upgrade, all other Event Management jobs wait for the platform upgrade to finish.

Note: During an Event Management plugin upgrade, ALL Event Management jobs do NOT work.
**Domain separation and Event Management**

This is an overview of domain separation and Event Management. Domain separation enables you to separate data, processes, and administrative tasks into logical groupings called domains. You can then control several aspects of this separation, including which users can see and access data.

### Overview

**Support: Level 1**

Domain separation is supported in this application. Not all ServiceNow applications support domain separation; some include limitations on the data and administrative settings that can be domain separated. To learn more, see Application support for domain separation.

### How domain separation works in Event Management

When domains are separated in Event Management, users can only see and manage alerts and events in their own (tenant) domain.

A domain column is present for Event Management tables that are provided in the base system. The column displays the name of the domain to which the event or alert belongs.

**Note:**

If the domain column does not display in the list, click (Update Personalized List) and add the required column. In addition, you can add a column that displays the domain path.

For information about domain separation limitations in Event Management, see Configure Event Management domain separation.

**Quick start tests for Event Management**

Validate that Event Management still works after you make any configuration change such as apply an upgrade or develop an application. Copy and customize these quick start tests to pass when using your instance-specific data.

Event Management quick start tests require activating the Event Management plugin (com.glideapp.itom.snac).

<table>
<thead>
<tr>
<th>Test</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMSelfMonitoring</td>
<td>Track problems with the Event Management plugin after upgrade. To understand the details of issues found, open the Alert Console and search for alerts that failed, whose source is EMSelfMonitoring, and that were opened or reopened since the upgrade. The exact time for filtering can be found in the error message.</td>
</tr>
</tbody>
</table>

**Components installed with Event Management**

Activating the Event Management (com.glideapp.itom.snac) plugin adds several roles, scheduled jobs, and tables.
Roles installed with Event Management

Roles used by the Event Management application.

Event Management adds these roles.

<table>
<thead>
<tr>
<th>Role title (name)</th>
<th>Description</th>
<th>Contains roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Management Administrator</td>
<td>Has read and write access to all Event Management features to configure Event Management.</td>
<td></td>
</tr>
<tr>
<td>(evt_mgmt_admin)</td>
<td></td>
<td>• evt_mgmt_user</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• template_editor_global</td>
</tr>
<tr>
<td>Event Management Operator</td>
<td>In addition to the evt_mgmt_user permissions, can also activate operations on alerts such as acknowledge, close, open incident, and run remediations.</td>
<td>• evt_mgmt_user</td>
</tr>
<tr>
<td>(evt_mgmt_operator)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Event Management User</td>
<td>Has read access to all Event Management features. Has write access to alerts to manage the alert life. Has the itil role to be able to manage incidents that are created from alerts.</td>
<td>• itil</td>
</tr>
<tr>
<td>(evt_mgmt_user)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Event Management Integrator</td>
<td>Has create access to the Event (em_event) and Registered Nodes (em_registered_nodes) tables to integrate with external event sources.</td>
<td></td>
</tr>
<tr>
<td>(evt_mgmt_integration)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Scheduled jobs installed with Event Management

List of scheduled jobs that are provided with Event Management.

To review the list of scheduled jobs, navigate to **System Scheduler > Scheduled Jobs**. Event Management adds the following scheduled jobs.

<table>
<thead>
<tr>
<th>Scheduled job</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Management — Connector execution job</td>
<td>Compares current time with time when active connector instances were last run and sets relevant connectors to execute. Runs every 10 seconds.</td>
</tr>
<tr>
<td>Event Management — Handle Impact Stuck Service</td>
<td>Releases stuck services and marks them as requiring rebuilding in the Impact Changes table to rebuild the impact tree. Runs every 2 minutes, 31 seconds.</td>
</tr>
<tr>
<td>Event Management — Impact Calculator Trigger</td>
<td>Trigger the impact calculation. The Event Management dashboard and impact tree are refreshed using the calculated figures. Runs every 6 seconds.</td>
</tr>
<tr>
<td>Scheduled job</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Event Management — Impact Topology Consumer</td>
<td>Consumes topology changes and marks the related services as ‘require rebuilding’ in the Impact Changes table to rebuild their impact trees.</td>
</tr>
<tr>
<td></td>
<td>Runs every 19 seconds.</td>
</tr>
<tr>
<td>Event Management — Update stuck connectors</td>
<td>Release connector instances that are stuck.</td>
</tr>
<tr>
<td></td>
<td>Runs every 2 minutes.</td>
</tr>
<tr>
<td>Event Management — Alert Priority Queue</td>
<td>Calculate <em>alert priority</em>. Two Alert Priority Queue jobs are active and available and can be run multi-thread.</td>
</tr>
<tr>
<td></td>
<td>Runs every minute.</td>
</tr>
<tr>
<td>Event Management — auto close alerts</td>
<td>Alerts that are idle longer than 7 days (default time period) are closed. Modify the default using the <code>evt_mgmt.alert_auto_close_interval</code> property.</td>
</tr>
<tr>
<td></td>
<td>Runs every 10 minutes.</td>
</tr>
<tr>
<td>Event Management — Calculate Alert Priority Grouping</td>
<td>Runs and calculates the priority groups: urgent, high, moderate and low for the alerts according to the highest and lowest priority score in the system.</td>
</tr>
<tr>
<td></td>
<td>Runs every 30 minutes.</td>
</tr>
<tr>
<td>Event Management — Close Flapping Alerts</td>
<td>Close flapping alerts.</td>
</tr>
<tr>
<td></td>
<td>Runs every 5 minutes.</td>
</tr>
<tr>
<td>Event Management — Close Threshold Alerts</td>
<td>Close threshold alerts.</td>
</tr>
<tr>
<td></td>
<td>Runs every 2 minutes.</td>
</tr>
<tr>
<td></td>
<td>Runs every 11 seconds.</td>
</tr>
<tr>
<td>Event Management — Create/resolved Incidents by Alerts</td>
<td>Job to:</td>
</tr>
<tr>
<td></td>
<td>- Create incidents for alerts according to alert action rules.</td>
</tr>
<tr>
<td></td>
<td>- Update incidents according to alert state.</td>
</tr>
<tr>
<td></td>
<td>Runs every 11 seconds.</td>
</tr>
<tr>
<td>Event Management — Impact Tree Builder</td>
<td>Handles all services with changes from the <code>em_impact_changes</code> table and rebuilds their impact trees.</td>
</tr>
<tr>
<td></td>
<td>Runs every 11 seconds.</td>
</tr>
<tr>
<td>Event Management — Insert Health Monitor</td>
<td>Job to produce the <strong>ServiceNow Event Management</strong> application service.</td>
</tr>
<tr>
<td></td>
<td>Runs once every hour.</td>
</tr>
<tr>
<td>Event Management — Maintenance Calculator</td>
<td>Calculate the maintenance for CIs.</td>
</tr>
<tr>
<td></td>
<td>Runs every minute.</td>
</tr>
<tr>
<td>Scheduled job</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Event Management — Node Count</td>
<td>Calculate <em>license usage</em>.</td>
</tr>
<tr>
<td></td>
<td>Runs once every hour.</td>
</tr>
<tr>
<td>Event Management — Process Events</td>
<td>Job that runs and processes each Ready event (apply event rule, mapping rule, and create or update alert)</td>
</tr>
<tr>
<td></td>
<td>Runs every 5 seconds.</td>
</tr>
<tr>
<td></td>
<td>Runs every 5 seconds.</td>
</tr>
<tr>
<td>Event Management — Queue Connector Processor</td>
<td>Bi-directional functionality. Processes all pending alerts in the Update Queue and sends them to the MID Server. By default, this dequeue process is performed in batches of 1,000 alerts.</td>
</tr>
<tr>
<td></td>
<td>Runs every 30 seconds.</td>
</tr>
<tr>
<td>Event Management — Recalculate Impact for Groups</td>
<td>By default, this job is not active. Can be run on demand to correct the impact on service groups.</td>
</tr>
<tr>
<td></td>
<td>Runs on demand.</td>
</tr>
<tr>
<td>Event Management — Recover Stuck Events</td>
<td>Handle all events that are in queued state and switch back to Ready to handle events from the beginning.</td>
</tr>
<tr>
<td></td>
<td>Runs at system startup.</td>
</tr>
<tr>
<td>Event Management — Update Health Monitor</td>
<td>Update the <em>ServiceNow Event Management</em> application service.</td>
</tr>
<tr>
<td></td>
<td>Runs once every hour.</td>
</tr>
<tr>
<td>Event Management — Update SLA Configuration Result</td>
<td>Synchronizes the CIs that match the SLA configuration filter with the Event Management SLA (em_ci_severity_task) table.</td>
</tr>
<tr>
<td></td>
<td>Runs every 10 minutes.</td>
</tr>
<tr>
<td>Event Management — Update SLA Severity</td>
<td>Updates Event Management SLA (em_ci_severity_task) table with the new severity.</td>
</tr>
<tr>
<td></td>
<td>Runs once every hour.</td>
</tr>
<tr>
<td>Event Management — Convert IT service</td>
<td>Run this property on demand to convert manual services to application services.</td>
</tr>
<tr>
<td></td>
<td>Runs every 30 minutes.</td>
</tr>
<tr>
<td>Event Management — Collect xmlstats</td>
<td>Collect event processing statistics.</td>
</tr>
<tr>
<td></td>
<td>Runs once every minute.</td>
</tr>
<tr>
<td>Event Management — Impact Calculator for Alert Groups and SLA</td>
<td>Calculates the effect of alerts on alert group services.</td>
</tr>
<tr>
<td>4x Event Management — Impact Calculator for Services</td>
<td>Calculates the impact of alerts on application services, and builds the impact tree on the Event Management dashboard. Four separate jobs run to ease the load on the system.</td>
</tr>
<tr>
<td>Event Management — Backfill Alert History table</td>
<td>Locates redundant records in the em_backfill_alert_history table and cancels them.</td>
</tr>
<tr>
<td>Scheduled job</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Event Management — Backfill Impact Status table</td>
<td>Locates redundant records in the <code>em_backfill_impact_status</code> table and cancels them.</td>
</tr>
<tr>
<td>Event Management — Impact for Groups</td>
<td>Calculates the impact of alerts on service groups.</td>
</tr>
<tr>
<td>Event Management — Clean Alert History Table</td>
<td>Cleans the Alert History (<code>em_alert_history</code>) table by removing records more than 90 days old. You can customize the amount of time after which alerts are removed by configuring <code>evt_mgmt.impact_calculation.cleanup_age_seconds.em_alert_history</code> in the <code>sys_properties.list</code> table.</td>
</tr>
<tr>
<td>Event Management — Clean Impact Status Table</td>
<td>Cleans the Impact Status (<code>em_impact_status</code>) table by removing records more than 90 days old. You can customize the amount of time after which alerts are removed by configuring <code>evt_mgmt.impact_calculation.cleanup_age_seconds.em_impact_status</code> in the <code>sys_properties.list</code> table.</td>
</tr>
</tbody>
</table>

Event Management adds the following scheduled jobs to support alert aggregation and RCA.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Analytics Purge Old Observation Data — Daily</td>
<td>Cleans the staging data.</td>
</tr>
<tr>
<td>Service Analytics Prepare RCA Learner Input Data — Daily</td>
<td>Prepares RCA input data. Stores and probes MID server to learn statistical information about alerts.</td>
</tr>
<tr>
<td>Service Analytics group alerts using RCA/Alert Aggregation</td>
<td>Applies RCA and alert aggregation to open alerts and prepares automated alert groups.</td>
</tr>
<tr>
<td>Service Analytics Alert Aggregation Learner — Daily</td>
<td>Learns information about existing alerts and groups new open alerts.</td>
</tr>
<tr>
<td>Service Analytics RCA Configuration</td>
<td>Configures root cause analysis.</td>
</tr>
<tr>
<td>Service Analytics Check File System Space on Analytics MID — Daily</td>
<td>Checks disk usage on the dedicated MID Server, and generates an event if it exceeds the threshold set in the <code>sa_analytics.rca.mid_max_allowed_space</code> property.</td>
</tr>
<tr>
<td>Service Analytics — Update virtual alerts for aggregation groups</td>
<td>Update the virtual alerts that were created to represent alert aggregation groups, with any changes to alerts belonging to that group. Runs every minute.</td>
</tr>
<tr>
<td>Service Analytics Attribute Populator for Historical Alerts</td>
<td>Populate attributes used in feature identifier for historical alert data using event rules. Runs on demand.</td>
</tr>
</tbody>
</table>

Tables installed with Event Management
Tables that are provided when Event Management is activated.

Event Management adds these tables.
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert (em_alert)</td>
<td>Alerts that Event Management manage.</td>
</tr>
<tr>
<td>Alert Correlation Rule (em_alert_correlation_rule)</td>
<td>Rules specifying primary and secondary correlated alerts.</td>
</tr>
<tr>
<td>Alert Aggregation Group Alerts (em_agg_group_alert)</td>
<td>Stores alerts associated with aggregated alert groups.</td>
</tr>
<tr>
<td>Alert Aggregation Group (em_agg_group)</td>
<td>Relationships between aggregated groups and primary alerts.</td>
</tr>
<tr>
<td>Alerts History (em_alert_history)</td>
<td>History of alerts. Used for impact calculation.</td>
</tr>
<tr>
<td>Alert Rule (em_alert_rule)</td>
<td>Mappings of alert fields to the Incident (incident) table.</td>
</tr>
<tr>
<td>Alert Template (em_alert_template)</td>
<td>Alert templates. This table extends the Template (sys_template) table.</td>
</tr>
<tr>
<td>Event Management SLA (em_ci_severity_task)</td>
<td>Event Management SLA tasks for CIs and business services.</td>
</tr>
<tr>
<td>Connector Definition (em_connector_definition)</td>
<td>Settings for gathering events from external event sources.</td>
</tr>
<tr>
<td>Connector Instance (em_connector_instance)</td>
<td>Connection details for external event sources.</td>
</tr>
<tr>
<td>MID Server to Connector Instance (em_connector_instance_to_mid)</td>
<td>Mappings of MID Servers to connector instances.</td>
</tr>
<tr>
<td>Event (em_event)</td>
<td>Events received by Event Management.</td>
</tr>
<tr>
<td>Event Filter (em_event_filter)</td>
<td>Storage for defined event filters.</td>
</tr>
<tr>
<td>Event Match Rule (em_match_rule)</td>
<td>Updated events for alert processing. Used by event rules.</td>
</tr>
<tr>
<td>Event Match Field (em_match_field)</td>
<td>Mappings of event fields to alert fields. Simple mapping. Used by Event Rules.</td>
</tr>
<tr>
<td>Event Mapping Rule (em_mapping_rule)</td>
<td>Updated event fields for alert processing.</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Event Type</td>
<td>Event types.</td>
</tr>
<tr>
<td>(em_event_type)</td>
<td></td>
</tr>
<tr>
<td>Task Template</td>
<td>Templates that define how to populate new tasks. For example, how fields of an incident that is being created from an alert, must be populated. This table extends the Template (sys_template) table.</td>
</tr>
<tr>
<td>(em_incident_template)</td>
<td></td>
</tr>
<tr>
<td>Registered Nodes</td>
<td>Registered nodes data.</td>
</tr>
<tr>
<td>(em_registered_nodes)</td>
<td></td>
</tr>
<tr>
<td>Threshold Rule</td>
<td>Alert threshold rules.</td>
</tr>
<tr>
<td>(em_threshold_rule)</td>
<td></td>
</tr>
<tr>
<td>Binding Device Map</td>
<td>Event binding to network paths and storage paths.</td>
</tr>
<tr>
<td>(Em_binding_device_map)</td>
<td></td>
</tr>
<tr>
<td>Process to CI Type Mappings</td>
<td>Event binding to specific processes.</td>
</tr>
<tr>
<td>(Em_binding_process_map)</td>
<td></td>
</tr>
<tr>
<td>CI Remediation</td>
<td>Remediation rule definitions.</td>
</tr>
<tr>
<td>(em_ci_remediation)</td>
<td></td>
</tr>
<tr>
<td>Impact Graph</td>
<td>Impact tree of CIs containing CI hierarchy and impact rules to be used for impact calculation.</td>
</tr>
<tr>
<td>(em_impact_graph)</td>
<td></td>
</tr>
<tr>
<td>Impact Graph History</td>
<td>History of changes in impact tree.</td>
</tr>
<tr>
<td>(em_impact_graph_history)</td>
<td></td>
</tr>
<tr>
<td>Impact Rule Definitions</td>
<td>Definition of rules used for impact calculation.</td>
</tr>
<tr>
<td>(em_impact_rule_definition)</td>
<td></td>
</tr>
<tr>
<td>Impact Rule instance</td>
<td>Rules based on impact rule definitions.</td>
</tr>
<tr>
<td>(em_impact_rule)</td>
<td></td>
</tr>
<tr>
<td>Infrastructure Relations</td>
<td>Child-parent pairs or CI types. CIs matching these definitions are added to impact trees.</td>
</tr>
<tr>
<td>(em_impact_infra_rel_def)</td>
<td></td>
</tr>
<tr>
<td>Impact Maintenance CIs</td>
<td>CIs that are in maintenance and therefore are excluded from impact calculation.</td>
</tr>
<tr>
<td>(em_impact_maint_ci)</td>
<td></td>
</tr>
<tr>
<td>Impact Status</td>
<td>Calculated status of CIs and services to be displayed in the dashboard and business service maps for technical services.</td>
</tr>
<tr>
<td>(em_impact_status)</td>
<td></td>
</tr>
<tr>
<td>SLA Configuration</td>
<td>SLA configuration records that identify the CIs that SLAs can run.</td>
</tr>
<tr>
<td>(em_sla_configuration)</td>
<td></td>
</tr>
<tr>
<td>Service Analytics Metric Type Registration</td>
<td>Source registration details for processing raw data.</td>
</tr>
<tr>
<td>(sa_metric_registration)</td>
<td></td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Application service (cmdb_ci_service_manual)</td>
<td>Stores records that represent Business Services that were created manually using Event Management &gt; Services &gt; Application Services capabilities, or imported from the Service (cmdb_ci_service) table. The added functionality of the Business Service table (cmdb_ci_service_manual) is that it supports Business Service maps and impact calculations.</td>
</tr>
</tbody>
</table>
| Health monitor scripts (em_monitor_scripts) | These scripts determine how to monitor or check, for example, when using the Connectors Monitor script. You can create customized script to monitor a device or an entity. The scripts provided with the base instance are:  
- Check delay in event processing  
- Connectors Monitor  
- Get Event Processing state  
- MID Server Threshold Alerts |
| Monitoring configuration (em_monitor_conf) | Use this table to configure what to monitor according to the scripts that are listed. Configure how often to run each script. If a script has a threshold, it determines what alert severity to display. Threshold values are in units of minutes and specify the delay time. Navigate to Event Management > Settings > Self-Health configuration to see the list of Monitoring Configurations or to create a new one. Use this script to test Data Center Monitoring.  
The scripts provided with the base instance are:  
- Connector’s idle state monitoring-monitor to verify whether any of the connectors was in idle state that surpassed the threshold (in minutes) that was configured.  
- Connectors Status- monitor to track the active status of the connectors.  
- Delay in event processing-monitor to track the duration (in minutes) of events that remained in ‘ready’ state and were not processed.  
- Event Processing job-monitors the state of the event processing jobs.  
- MID Server Threshold Alert-monitors MID Server health. |
| Monitoring state (em_monitor_state) | Use this table to set the threshold for each connector. When there is a value above the threshold, an alert is generated. A business service displays the status, for example, in the Event Management dashboard or Alert Console. |
| EM XMLStats Data (em_xmlstats_data) | Self-health statistics and diagnostic details for Operational Intelligence and Event Management, which are used to produce the XMLStats page. |

Event Management adds the following tables to support alert aggregation and RCA.
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA RCA Status (sa_rca_status)</td>
<td>Information (such as IDs) for the latest messages that were sent to the ECC Queue for a service during RCA.</td>
</tr>
<tr>
<td>SA RCA Output (sa_rca_output)</td>
<td>RCA learner output data.</td>
</tr>
<tr>
<td>SA RCA Group (sa_rca_group)</td>
<td>Automated alert groups for the RCA query.</td>
</tr>
<tr>
<td>SA Analytics Alert Staging (sa_analytics_alert)</td>
<td>Staging table for alerts used for analytics.</td>
</tr>
<tr>
<td>SA RCA Input (sa_rca_input)</td>
<td>Input data for the RCA learner.</td>
</tr>
<tr>
<td>SA Analytics Status (sa_analytics_status)</td>
<td>Last run information to be used for alert aggregation and RCA.</td>
</tr>
<tr>
<td>SA RCA Group Alert (sa_rca_group_alert)</td>
<td>Alerts associated with automated alert groups.</td>
</tr>
<tr>
<td>SA RCA Service Configuration Item Association (sa_rca_svc_ci_assoc)</td>
<td>Associations between CIs and services.</td>
</tr>
<tr>
<td>SA RCA SMC Config Base (sa_rca_smc_config_base)</td>
<td>State Model Configuration base. User defined RCA configurations. Each configuration is associated with one or more rules in the SA RCA SMC Rule Base (sa_rca_smc_rule_base) table, if applies.</td>
</tr>
<tr>
<td>SA RCA SMC Rule Base (sa_rca_smc_rule_base)</td>
<td>Service Analytics (SA) Root Cause Analysis (RCA) State Model. Individual rules that are associated with RCA configuration in the SA RCA SMC Config Base (sa_rca_smc_config_base) table.</td>
</tr>
<tr>
<td>SA RCA SMC Config (sa_rca_smc_config)</td>
<td>RCA Configuration revisions table. Snapshots of RCA configurations generated during configuration comparisons.</td>
</tr>
<tr>
<td>SA RCA SMC Deployment (sa_rca_smc_deployment)</td>
<td>Information about the current revision of the RCA configuration that is in effect, and the RCA configuration that is set to be deployed at the next daily run cycle of the Learner.</td>
</tr>
<tr>
<td>SA RCA SMC Run (sa_rca_smc_run)</td>
<td>RCA SMC (State Model Configuration) Run table. All comparisons between two RCA configurations that the user ran.</td>
</tr>
</tbody>
</table>
### Table

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA Alert Aggregation Learned Pattern (sa_agg_pattern)</td>
<td>Learned patterns from alert aggregation.</td>
</tr>
<tr>
<td>SA Alert Aggregation Learned Pattern Elements (sa_agg_pattern_element)</td>
<td>CI/Metric Name pairs associated with learned patterns.</td>
</tr>
<tr>
<td>SA Alert Aggregation Query Group Patterns (sa_agg_group_pattern)</td>
<td>Relationships between groups discovered in alert aggregation queries and patterns found in learning.</td>
</tr>
<tr>
<td>SA Alert Aggregation Query -- Staged (Recent) Alerts (sa_agg_group_alert_staging)</td>
<td>A staging table for alerts that have not yet been associated with any aggregated alert group.</td>
</tr>
<tr>
<td>SA Agg Pattern Attribute (sa_agg_pattern_attribute table)</td>
<td>CI/alert attributes to be used for finding patterns for alert aggregation.</td>
</tr>
<tr>
<td>SA Alert Attribute Populator Status (sa_alert_attribute_populator_status_table)</td>
<td>State and statistics for attribute populator job.</td>
</tr>
<tr>
<td>SA Alert Aggregation Learned Pattern Elements Pair wise Mutual Information and Joint Probability (sa_agg_pattern_element_pair)</td>
<td>Pairwise probabilities for pattern elements.</td>
</tr>
<tr>
<td>EM Agg Group Prediction (em_agg_group_prediction)</td>
<td>Alert predictions for alert groups.</td>
</tr>
</tbody>
</table>

### SLAs for business services and CIs

Event Management supports the creation of SLAs for business services and for CIs.

You can monitor and manage the quality of the business services offered by your organization. For example, you might want to set up an SLA that is triggered when the max alert severity of a business service or CI is **Critical**, and measure the time it takes to bring the business service or CI back to the **Info** state, when the conditions that caused the alert have been resolved.

### SLAs and the Task table

You can create SLA definitions only for tables that extend the Task table. The Event Management application provides a table named Event Management SLA (em_ci_severity_task), which extends the Task table. Use this table in your SLA definitions to specify the severity level that should trigger and stop the SLA. During alert impact calculation, changes in the severity level of business services and CIs are automatically updated in the Event Management SLA table. **Scheduled jobs** keep the information in this table up to date.

### How the Event Management SLA table is populated

The Event Management SLA table is populated differently for business services and CIs:
For business services, the system automatically populates the Event Management SLA table when a business service is created or when its max severity is changed.

For CIs, you must first identify which CIs can be made available for SLAs by creating an SLA configuration record. The system then automatically populates the Event Management SLA table with the CI max severity is changed.

**Note:** Duplicate CIs are not added to the Event Management SLA table even if the same CI matches more than one SLA configuration filter.

Create an SLA configuration for CIs

Create an SLA configuration from the Event Management application to determine which CIs are available for SLAs.

Role required: evt_mgmt_admin

The SLA configuration record is a filter on a table in the system. Select a table that has the CIs for which you want to create an SLA. If you are creating an SLA definition on a business service, you do not need to create an SLA configuration.

1. Navigate to **Event Management > Settings > SLA Configuration**.
2. Click **New**.
3. Fill in the form fields (see table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive name.</td>
</tr>
<tr>
<td>Table</td>
<td>Select a table with the CIs for which you want to create the SLA. You can also use the Configuration Item to Host View (query_based_service_basic_view) table, which is used by <strong>technical services</strong>.</td>
</tr>
<tr>
<td>Filter</td>
<td>Configure the filter for the records.</td>
</tr>
</tbody>
</table>

4. Right-click the header and select **Save**.

The following is an example.

If you want to run an SLA on all Linux servers for major and critical alerts, first create an SLA configuration on the Linux Server (cmdb_ci_linux_server) table. Then create the SLA definition with this filter:

- **CI.class** $Linux Server
- **Severity** $Major Or $Critical

**Create an SLA definition** on the CIs that match this SLA configuration.

Limit the records for the SLA configuration filter

If too many records are returned by the SLA configuration filter, you can add a property to set the maximum number of records.

Role required: evt_mgmt_admin

By default, the size of technical service records is limited to 10,000 elements. The result view is truncated beyond this limit. If the filter configuration for a specified technical service returns too
many records, you can limit the number of records that are returned by setting the value of the `sa.qbs.max_num_of_cis` property. For more information, see Create a technical service.

1. Enter `sys_properties.list` in the application navigator.
2. Add the following property:
   
   `sa.qbs.max_num_of_cis`
3. Set the value to an integer that represents the maximum number of records to return.

Create an SLA definition for a CI or business service
You can create SLA definitions for CIs and business services just as you can for other task records in the instance.

Role required: `evt_mgmt_admin`

1. Navigate to Service Level Management > SLA > SLA Definitions.
2. Click New.
3. Fill out the fields on the SLA Definition form. For a description of each field, see .
4. For the Table field, select Event Management SLA (`em_ci_severity_task`).
5. Configure the Start condition filter by adding conditions, such as:

<table>
<thead>
<tr>
<th>SLA filter conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>For all CIs where the Severity is Critical</strong></td>
</tr>
<tr>
<td><strong>Type</strong> is <code>Critical</code></td>
</tr>
<tr>
<td><strong>Severity is Critical</strong></td>
</tr>
<tr>
<td><strong>Business Service is Email</strong></td>
</tr>
<tr>
<td><strong>Severity is Critical</strong></td>
</tr>
</tbody>
</table>

6. Also configure pause, stop and reset conditions.
7. Click Submit.

Event Management configuration preferences
Preferred settings of properties and general configuration.

Use the Knowledge Base on HI and the Community to further help you find information issues.

General preferences

Self-health

By default, the self-health monitoring feature is not enabled. To enable it, navigate to Event Management > Settings > Properties and select Yes for the Enable Event Management self-health monitoring (`evt_mgmt.self_health_active`) property. Use this feature to monitor and track many Event Management features.
Use the following settings to help with preventing performance degradation.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Details</th>
</tr>
</thead>
</table>
| **Business rules** |  - Avoid writing business rules for event (em_event) tables, as they do not run in the current default REST URL that is used for event injection.  
  - Business rules that are written for alert (em_alert) tables must be highly efficient or they may result in performance degradation. Instead of writing a business rule, consider whether it is more appropriate to write a job. An inefficient business rule can cause incident creation for an alert to fail and the alert impact calculation to fail.  
  - Do not write async business rules for alert tables.  
  - Business rules must not change the **Category** field on event (em_event) tables. |
| **Scaling up** | Check the average event processing time before scaling up event throughput when first starting with Event Management. Do this check after an initial flow of events and all rules are in place. If processing time takes over a few milliseconds per event, determine the cause for the processing slowdown before continuing to scale. Performance duration can be checked in the Performance Statistics (sa_performance_statistics) table. |
| **Configure for large-scale environments** |  - Set the `Enable multi node event processing` (**event_processor_enable_multi_node**) property to **Yes**.  
  Enable multi-node in production environments and set values based on the size of the deployment and expected event rate.  
  - Set the Number of events to process in bulk during each event processing job (**evt_mgmt.max_objs_in_alert_query**) property to 500.  
  - Set the Number of scheduled jobs processing events (**event_processor_job_count**) property to 4.  
  - Set the Maximum events to be processed by every scheduled job (**event_mgmt.max_events_processing_per_job**) property to 5000.  
  - If you are sending events from a custom source, verify that events have **Message Key** or **Source**, **Node**, **Type**, and **Resource** data. |
| **Latency issues for receiving events** | Check the following settings:  
  - Verify that the **Bucket** field in the event (em_event) table is set to a value that is greater than zero (**0**).  
  - Navigate to **System Scheduler > Scheduled Jobs** and search for - **process events**.  
  Check that all - **process events** jobs exist according to the Number of scheduled jobs processing events (**event_processor_job_count**) property configuration. Verify that the State is **Running** or **Ready**. If the state is **Queued** or **Error**, set the job state to **Ready**. |
| **Archive events** |  - Avoid changing the default retention time for events.  
  - To log events for a longer time, create an archive table and a job that copies new events to it. Do this by scheduling a job to regularly back up events (em_event) to a custom table.  
  - Do not extend table rotation by adding more days. |
Event integration

SNMP traps

- Use a monitoring tool to send SNMP traps, rather than sending them directly from devices.
- To avoid having to rewrite event rules, upload MIBs prior to defining the event rules.

Web service API

- Using a web service API for integration can reduce the number of event rules needed. This action avoids having to transform events (prepared data is sent in an event to the instance).
- Use dedicated credentials for integration. Optionally, designate credentials specific to each event source.

CloudWatch

Use dedicated credentials for integrating CloudWatch with ServiceNow.

Email

Use email only if the source has a low volume and other options are not available, such as, running a script or forwarding an SNMP trap.

Event rules

Configuration settings when creating event rules:

- Write Event Rules to apply to the broadest number of events possible. More specific rules can then be created as necessary and should use a lower-order value.
- If a more general rule can achieve the same outcome, avoid writing Event Rules that apply only to a certain subset of events.
- When Event Rules are applied to events, no changes are made to the original event. All processing occurs in memory, so use the Processing Notes field and/or use the Check Process of Event UI action link to troubleshoot.
- If you change a rule/transform that has existing mapping rules, you should review and retest with events that are either actual or simulated.
- Ensure that the From field value exactly matches a string in the JSON in the additional_info field of an event. This matching happens when a rule has been configured based on information in a MIB file. If the MIB file is not uploaded, the JSON for the SNMP trap shows varbinds (variable binding) with dotted names, instead of the translated name in the MIB. The event field mapping rule then fails to be applied.
- Establish a consistent naming convention. A common convention is: <customer acronym>.<Event Source>.<Description>. For example, ACME.OEM.Normalize
- If two Event Rules have similar conditions set, use the Order field to control which Event Rule runs.
- Use Event Rules to associate an alert with a CI.

Additional settings for constructing event rules:

<table>
<thead>
<tr>
<th>Desired result</th>
<th>Required activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effective de-duplication and enabling efficient parallel event processing</td>
<td>Populate the Source, Node, Type, Resource, Metric Name fields.</td>
</tr>
<tr>
<td>Desired result</td>
<td>Required activity</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CI binding</td>
<td>• Bind to host - by populating the Node field and optionally CI identifiers.</td>
</tr>
<tr>
<td></td>
<td>• Binding to application and / or device – by populating the CI identifiers field and the Additional information field.</td>
</tr>
<tr>
<td>Alert Correlation, using alert aggregation and RCA</td>
<td>Populate the Resource and Metric Name fields.</td>
</tr>
<tr>
<td></td>
<td>Note: If CI is also bound, Alert Correlation is improved.</td>
</tr>
</tbody>
</table>

**Custom event fields**

Include additional fields in the Additional information field of the event only.

Do not add additional fields to an event by adding a custom field to the event (em_event) table.

Do not add columns to the event (em_event) table.

For information about how to include additional fields in events, see [Custom alert fields](#).

**De-Duplication**

Configuration settings for de-duplication.

- The message_key field is used for De-Duplication. If reliable message key values are not provided with the source event, it is important to have a well-defined plan for constructing these identifiers.
- If the message key is not defined, then the default message key is `<Source + Node + Type + Resource + Metric Name>`.
- The guideline is to have the event source populate the `<Source + Node + Type + Resource + Metric Name>` fields out-of-the-box and populate the message key. This action enables a better distribution of event processing among instance workers and nodes.
- If the source event does not have values for these fields, make sure to populate them using transform rules. This action does not affect event processing, but is used for de-duplication. Populate as many of these fields as possible before they are sent to the instance. This action provides better distribution of events over the processor workers, and therefore better throughput and scale.

**CI binding**

- Where possible, always attempt to bind an alert to a CI.

  **Note:** Although Event Rules are defined on events, CIs are bound to alerts that result from those events, and are not bound to the event.

- To bind a host, machine or any device with an IP, populate the event Node field with a unique host name, FQDN, IP, or MAC address. If other identifiers are necessary to identify a host, then populate the ci_identifiers field with a JSON format. The JSON format must contain the CMDB field name and value to perform the match.

  **Note:** The event Node field must be populated from an event rule or populated with a unique host name from the source before the event is inserted.
The primary binding strategy is to use the **Node** field. If the **Node** field is not pre-populated in the event, it can be populated using event rules.

**Alert settings**

**Alert lifecycle**

General alert functionality:

- An alert is opened whenever an event is not ignored or its threshold is exceeded by an event rule, and de-duplication does not identify the event as belonging to an existing alert.
- An alert is closed when a closing event is sent on the same message key, or the alert is closed manually.
- An alert is reopened if an opening alert that has the same message key is sent within the timeframe defined in properties (default is one hour).
- If an alert is opened and closed at a high rate, as defined in properties, it becomes flapping. When this opening and closing rate stops, the alert goes out of flapping state.
- If an incident is opened from an alert, that alert remains open as long as the incident remains open. By default, when either the incident or the alert is closed the other is closed as well. This behavior can be configured using properties.
- Do not close an alert when creating a corresponding incident.
- Do not delete an open alert. Close an alert first and then delete it.
- Use **Acknowledge** to denote that the alert is known, and can temporarily be ignored.
- Do not use **Acknowledge** to mark an alert as needing attention.
- Do not create alerts in any of these states:
  - Closed
  - Info
  - Open

The **evt_mgmt.alert_auto_close_interval** property automatically closes alerts after the specified period. Do not specify 0, as this value disables the feature and may lead to performance degradation.

Do not create alerts in **Info** state. In some monitoring systems **Info** denotes that an issue has been resolved, while in other monitoring systems **Info** is used to denote events that are not of operational significance. For the former case, use **Clear** instead of **Info** using a Mapping Rule. For the latter case, have an Ignore rule, unless the events are of specific value.

**Alert action rules**

- A scheduled job applies Alert action rules to new Alerts every 1 1 seconds. If an Alert Rule does not immediately start, allow 10–15 seconds before you start troubleshooting.
- Use the **Order** field to control which Alert Rule runs if two Alert Rules have similar conditions set.
- Use Alert action rules with Task Templates to populate static values in an incident. Use the populator script to assign dynamic values in the incident. The populator script can return a value of false to abort incident creation.
- Create a user called Event Management (or a similar name). Then the **Created by** field in a task template (for example, **Incident**) can be set to indicate that user was the source of the task.
- To perform any dynamic value assignment or to override OOB dynamic value assignment, use the **EvtMgmtCustomIncidentPopulator** script include.
Remediation

- Always set orchestration workflow properties to the Remediation Task (em_remediation_task) table.
- Use ECC Queue and Workflow > Live Workflow > All Contexts to find more detailed information on remediation activities.

Business rules

- Business rules created on alert tables should not take more than a few milliseconds. In place of using a business rule, consider if the same functionality can be achieved using a job.
- Do not use business rules to associate an alert with a CI. Use event rules to do binding instead of using business rules.

Planning

- Organize event source configuration of filters, modules, and so on, into multiple parallel efforts, rather than in serial.
- Validate processed event formats to ensure that data that is parsed is aligned with desired results.
- Test production events in a non-production environment. Integrate with non-production element managers and ServiceNow instances. If non-production element managers are not available, send events from element managers to both production and non-production environments.

Services and dashboard

- Use Service Groups to group business services into logical groups to reduce the number of services displayed on the Service Health dashboard.
- Import manually built service maps.

Operational Intelligence collector logs and files

Operational Intelligence collector logs and files are located under the path $ (MID_SERVER_DIR)/agent. Use these logs and files for troubleshooting and monitoring purposes.

<table>
<thead>
<tr>
<th>Log or file</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerShell metric collector log file</td>
<td>Logs/retrieve_metrics(connector instance ID).log</td>
</tr>
<tr>
<td>PowerShell output file</td>
<td>work/metrics/metrics_output_{connector instance ID}.txt</td>
</tr>
<tr>
<td>PowerShell input file</td>
<td>work/metrics/parameters_{connector instance ID}.txt</td>
</tr>
</tbody>
</table>
Operational Intelligence performance can be checked in the MID Server log file when the `mid.log.level` MID Server parameter is in debug mode.

Operational Intelligence performance numbers are available in the Performance Statistics (sa_performance_statistics) table. To view the performance numbers, filter the Performance Statistics list for Metric Collector.

**Additional Event Management setup tasks**

After setting up and configuring Event Management to receive and process events and generate alerts, you can perform additional setup activities.

*Create a technical service*

A technical service is a dynamic grouping of configuration items (CIs), based on some common criteria. For example, you can create a technical service based on the location of all web servers in Detroit or all Oracle databases in Boston.

Role required: evt_mgmt_admin

Use the technical services form to create user defined collections of CIs.

**Note:** A technical service contains CIs. However, a technical service cannot contain other services.

This video explains Event Management basics, including showing how to create a technical service.

1. Navigate to Event Management > Services > Technical Services.

![Technical Services form](image)

2. Click New.

3. On the form, fill in the fields.

**Technical Service form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The business service name.</td>
</tr>
</tbody>
</table>
| Business criticality | The importance of this service to the business. This field can determine disaster recovery strategies for this service.  
|                   | • 1 - most critical  
|                   | • 2 - somewhat critical  
|                   | • 3 - less critical  
<p>|                   | • 4 - not critical |
| Owned by         | The party that is responsible for the business service.                    |
| Email            | Enter the email address of the owner who is responsible for this business service in the organization. |
| Business phone   | Enter the business phone number of the owner who is responsible for this business service in the organization. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational status</td>
<td>The status of the business service.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Operational</strong>: The business service is running. Impact calculation runs only on services in the Operational status.</td>
</tr>
<tr>
<td></td>
<td>Select this option to use this business service. If you customized preconfigured options for this category, select the option whose value is set to 1. Event Management calculates impact only on business services in use.</td>
</tr>
<tr>
<td></td>
<td>To use this business service at a later time, select one of these options:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Non-Operational</strong>: The business service is unavailable, disabled, or down.</td>
</tr>
<tr>
<td></td>
<td>- <strong>DR Standby</strong>: The business service is available as a disaster recovery (DR) standby.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Repair in Progress</strong>: The business service is receiving maintenance.</td>
</tr>
<tr>
<td>Comments</td>
<td>Any additional notes and information.</td>
</tr>
</tbody>
</table>

4. Select a source table and specify the criteria for the technical service query.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>Name of source table to be searched.</td>
</tr>
<tr>
<td></td>
<td>You can use autocomplete to locate table names, or select the appropriate table from the list.</td>
</tr>
<tr>
<td>Filter</td>
<td>Specify the criteria for determining the values from the specified source table to be displayed:</td>
</tr>
<tr>
<td></td>
<td>1. Specify the filter condition.</td>
</tr>
<tr>
<td></td>
<td>2. Choose the relevant operator.</td>
</tr>
<tr>
<td></td>
<td>3. Choose the input field.</td>
</tr>
<tr>
<td></td>
<td>4. Choose the operator.</td>
</tr>
<tr>
<td></td>
<td>5. Choose the input value.</td>
</tr>
<tr>
<td></td>
<td>If the filter configuration returns too many records, you can limit the number of records that are returned by setting the value of the sa.qbs.max_num_of_cis property. By default, the technical service size is limited to 10,000 elements and the result view is truncated.</td>
</tr>
</tbody>
</table>

5. Click Submit.

*Convert manual services to application services*

You can convert existing manual services to application services. Event Management can use application services to monitor service performance and identify health issues.

Role required: evt_mgmt_admin
Manual services can no longer be created and this service type, as well as discovered services, have been replaced by application services. Existing manual services continue to work and can be updated. However, new services that are created should be application services. Existing manual services should be converted to application services.

During the conversion process:

- The manual service record is moved from the Manual Services (cmdb_ci_service_manual) table to the Application Services (cmdb_ci_service_discovered) table by changing the record class.
- All the original manual service attributes, such as, name, owner, and operational status, are retained in the application service.
- The configuration items (CIs) of the manual service are retained in the application service after the conversion.
- The system queries the CMDB for the latest CI changes and applies CI impact rules.

Application service does not support these CI types:

- cmdb_ci_endpoint
- cmdb_ci_translation_rule
- cmdb_ci_config_file
- cmdb_ci_qualifier
- cmdb_ci_application_cluster

If the preceding CI types are used in the manual service that you want to convert, the conversion fails. CIs of this type should be removed before conversion. They can be replaced by adding new CIs to the application service after conversion.

If your ServiceNow instance uses domain separation, the conversion fails if CIs from multiple domains are in the same service. Replace or remove relevant CIs before conversion. The converted application service contains only CIs assigned to the same domain as the application service.


2. Select the manual service that you want to convert to an application service.
3. Click **Migrate to Application Service**.
4. Click **OK**.

- The CIs from the manual service are added to the converted application service.
- After the conversion process is complete, the application service form opens.
Convert manual services to application services using API

You can use a JavaScript API to convert existing manual services to application services. Event Management can use application services to monitor service performance and identify health issues.

Manual services can no longer be created and this service type, as well as discovered services, have been replaced by application services. Existing manual services continue to work and can be updated. However, new services that are created should be application services. Existing manual services should be converted to application services.

During the conversion process:

- The manual service record is moved from the Manual Services (cmdb_ci_service_manual) table to the Application Service (cmdb_ci_service_discovered) table by changing the record class.
- All the original manual service attributes, such as, name, owner, and operational status, are retained in the application service.
- The configuration items (CIs) of the manual service are retained in the application service after the conversion.
- The system queries the CMDB for the latest CI changes and applies CI impact rules.

Application service does not support these CI types:

- cmdb_ci_endpoint
- cmdb_ci_translation_rule
- cmdb_ci_config_file
- cmdb_ci_qualifier
- cmdb_ci_application_cluster

If the preceding CI types are used in the manual service that you want to convert, the conversion fails. CIs of this type should be removed before conversion. They can be replaced by adding new CIs to the application service after conversion.

If your ServiceNow instance uses domain separation, the conversion fails if CIs from multiple domains are in the same service. Replace or remove relevant CIs before conversion. The converted application service contains only CIs assigned to the same domain as the application service.

The JavaScript API is:

```javascript
BusinessServiceManager.migrateManualToApplicationService(String service_id)
```

<table>
<thead>
<tr>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
</tr>
<tr>
<td>service_id</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Returns</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td>Boolean</td>
</tr>
</tbody>
</table>
Where the `sys_id` of an existing manual service is 451047c6c0a8016400de0ae6df9b9d76, run the following API to convert it to an application service:

```javascript
var bsManager = new SNC.BusinessServiceManager();
var res = bsManager.migrateManualToApplicationService("451047c6c0a8016400de0ae6df9b9d76");
```

### Manually create an Event Management application service

You can manually create an application service. Event Management can use application services to monitor service performance and identify health issues.

**Role required:** evt_mgmt_admin

An application service is a set of interconnected applications and hosts which are configured to offer a service to the organization. Application services can be internal, like an organization email system or customer-facing, like an organization website. Use application services to organize, maintain, and monitor services in your organization.

Application Services do not support these CI types:
- cmdb_ci_endpoint
- cmdb_ci_translation_rule
- cmdb_ci_config_file
- cmdb_ci_qualifier
- cmdb_ci_application_cluster

1. Navigate to **Event Management > Services > Application Services**.
2. Click **New**.
3. On the form, fill in the fields.

#### Create a Service form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique application service name.</td>
</tr>
<tr>
<td>Owned by</td>
<td>Specify an owner who is responsible for this application service in the organization</td>
</tr>
</tbody>
</table>

**Add entry points:**

- **CI Type**
  - Select the CI type (CI class) for the CI you are adding. Every CI belongs to a CI type which contains a set of attributes configured for this kind of CI, for example, cmdb_ci_appl for applications.

- **CI Name**
  - Select the CI from the list of CIs of the selected CI type.

The CI type list includes only allowed CI types. For example, you cannot add an application cluster.

4. Click **Submit**.
Control user access to application services
Assign user roles to service groups to grant users access to application services in your organization. Your organization may restrict access to some services for security or secrecy reasons.

Make sure that you have performed the user provisioning tasks for the users you want to grant access:

1. Add users to user groups.
2. Create new roles.
3. Assign roles to users or user groups.

Also, make sure that you have created service groups as described in Group application services.

Role required: app_service_admin or sm_admin

In the base system, the following roles provide access to application services:

app_service_admin
Creates and modifies application services, creates service groups, views, and edits application service maps.

app_service_user
Views application service maps. The itil role that serves as the basic helpdesk technician role contains the app_service_user role.

Service Mapping provides these preconfigured roles:

sm_admin
Sets up the Service Mapping application. Maps, fixes, and maintains application services. Also performs advanced configuration and customization of the product. Assign this role to application administrators.

sm_user
Views maps for operational application services to plan change or migration, as well as analyze the continuity and availability of services. Assign this role to application users.

sm_app_owner
Provides information necessary for successful mapping of an application service. Once a service is mapped, this user reviews the results and either approves it or suggests changes. Assign the sm_app_owner role to users who own application services and are familiar with the infrastructure and applications that make up the services.

Event Management provides these preconfigured roles:

evt_mgmt_admin
Has read and write access to all Event Management features to configure Event Management.

evt_mgmt_operator
In addition to the evt_mgmt_user permissions, can also activate operations on alerts such as acknowledge, close, open incident, and run remediations.

evt_mgmt_user
Has read access to all Event Management features. Has write access to alerts to manage the alert life. Has the itil role to be able to manage incidents that are created from alerts.

**evt_mgmt_integration**

Has create access to the Event (em_event) and Registered Nodes (em_registered_nodes) tables to integrate with external event sources.

Typically, enterprises have hundreds of services which makes it impractical to manage them individually. Service groups can make service lists much shorter and easier to manage, especially in large organizations or service providers. In a hierarchy of service groups, access to a parent service group automatically grants access to all the child service groups.

Users inherit permissions from roles that are assigned to them. You can assign some roles directly to service groups to allow all users with this role to access all application services belonging to this group. However, most enterprises choose to organize their roles as a hierarchy. It helps to manage roles across multiple ServiceNow applications. For example, the Service Mapping administrator (sm_admin) can be part of a broader administrator role like administrator (admin). You can add users to user groups and then assign roles to the user groups to give permissions of this role simultaneously to all the group users.

**Assigning a role to an application service group**

By default, all services are assigned to the **All** service group that lets all users view and manage application services. When you assign a role to a service group, the users with this role can access only application services in this service group. To enable users with this role to access other services, assign this role to the respective service group.
1. Navigate to either of the following:
   - Configuration > Application Services > Service Group Responsibilities.
   - If Service Mapping is activated: Service Mapping > Services > Service Group Responsibilities.
   - If Event Management is activated: Event Management > Services > Service Group Responsibilities.

2. Click New and fill out the Business Service Group Responsibilities form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Service Group</td>
<td>Service group to which you want to assign a role.</td>
</tr>
<tr>
<td>Role</td>
<td>Role you want to assign to the selected service group.</td>
</tr>
<tr>
<td></td>
<td>For example, financial_services_admin.</td>
</tr>
</tbody>
</table>

3. Click Submit.

To manage access to services that contain sensitive financial information in your organization:
1. Organize the services into the Financial Services group.
2. Create a new user role, financial services administrator (financial_services_admin) role, that contains the (app_service_it) role.
3. Assign the Financial Services administrator role to the Financial Services group.

As a result, only users with the Financial Services administrator role can access application services belonging to the Financial Services group.

Group application services
Organize application services by groups to perform actions simultaneously on multiple services, and to control user access to services. You can use Event Management to track service health by service groups.

Role required: sm_admin or app_service_admin

Typically, enterprises have hundreds of services which makes it impractical to manage them individually. Service groups can make service lists much shorter and easier to manage, especially in large organizations or service providers.

How you group application services depends on the user and on service provisioning policies in your enterprise. The relation between application services in groups is purely logical and the same application service can belong to multiple groups. For example, the Mobile service can be part of the following service groups: sales, Beijing, and telephony.
You can embed a service group within another service group to create a hierarchy of service groups. If users have access to a parent service group, they automatically have access to all its child groups. By default, all services are assigned to the All service group that lets all users view and manage application services. When you assign a role to a service group, the users with this role can access only application services in this service group. To enable users with this role to access other services, assign this role to the respective service group.

If Service Mapping is activated, service groups can contain a mixture of manually created application services and application services discovered by Service Mapping.

1. Navigate to Configuration > Application Services > Service Groups. Or, if Service Mapping is activated, navigate to Service Mapping > Services > Service Groups.
2. Click New.
3. Enter the name of the new application service group in the Name field.
4. To embed this group in another group, enter the name of the other group in the Parent Group field.
5. Right-click the form header and click Save.
6. Add an application service to the newly created service group.
   a) In the Service Group Members section, click New.
   b) In the Name field, enter the name of the application service. If you are using Event Management, you can also enter an alert group name.
   c) Click Submit.
7. Alternatively, add an application service to a group from the application service form.
   a) Navigate to Configuration > Application Services > Application Services.
   b) Select the application service you want to add to a service group.
   c) In the Service Group Members section, double-click Insert a new row.
d) Enter the name of the service group to which you want to add the selected application service.

e) Click the OK icon.

f) Click Update.

Create an alert group

An alert group is a set of alerts that meet special criteria for a particular business service.

Navigate to Event Management > Services > Properties and ensure that the Enable alert group support (evt_mgmt.impact_calculation.alert_group_support) property is set to Yes.

Role required: evt_mgmt_admin or evt_mgmt_operator

The main motivation to use alert groups is a modeling solution based on data contained in the alert itself, as an alternative to using either discovered, application, or technical services.

Create alert groups to combine similar alerts that meet the specified criteria. Alert groups appear on the Event Management dashboard.

Note: Alert groups that are created in this procedure do not display in the Alerts console.

You can learn about Event Management basics, including alert groups, from this video:

1. Navigate to Event Management > Services > Alert Groups.
2. Click New
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The alert group name.</td>
</tr>
<tr>
<td>Business criticality</td>
<td>The importance of this service to the business. This field can determine disaster recovery strategies for this service.</td>
</tr>
<tr>
<td>Used for</td>
<td>Specify the purpose of the alert group. Select from:</td>
</tr>
<tr>
<td>Owned by</td>
<td>The party that is responsible for the business service.</td>
</tr>
<tr>
<td>Email</td>
<td>The email alias for the alert group.</td>
</tr>
<tr>
<td>Business phone</td>
<td>The phone number for the business.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Operational status</td>
<td>The status of the business service.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Operational</strong>: The business service is running. Impact calculation runs only on services in the Operational status.</td>
</tr>
<tr>
<td></td>
<td>Select this option to use this business service. If you customized preconfigured options for this category, select the option whose value is set to 1. Event Management calculates impact only on business services in use.</td>
</tr>
<tr>
<td></td>
<td>If you do not want to use this business service now, select one of these options:</td>
</tr>
<tr>
<td></td>
<td>· <strong>Non-Operational</strong>: The business service is unavailable, disabled, or down.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Repair in progress</strong>: The business service is receiving maintenance.</td>
</tr>
<tr>
<td></td>
<td>· <strong>DR Standby</strong>: The business service is available as a disaster recovery (DR) standby.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Ready</strong>: Prepared for use.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Retired</strong>: Withdrawn from active use.</td>
</tr>
<tr>
<td>Comments</td>
<td>Any additional notes and information.</td>
</tr>
<tr>
<td>Service Group Members</td>
<td>The business services group which contains this service.</td>
</tr>
<tr>
<td>Filter</td>
<td>The conditions for sending a notification to the alert group. The conditions must meet the alert values.</td>
</tr>
</tbody>
</table>

**Note:** In the **Filter** field:
- The filter that defines Alert Groups should not be on fields that do not appear in the `em_alert_history` table because impact calculation is not calculated properly. This situation occurs because fields like Event Count, Priority, and Priority Group are not copied to the `em_alert_history` table for impact calculation.
- Do not specify a dynamic time condition. For example, in the filter, do not specify "Created" condition of "Last 45 minutes" because impact calculation is triggered by a change of Alert or Alert group. However, for the dynamic time condition, none of these conditions have changed.

4. Click **Update**.

**Create a service group**
Create service groups to combine similar services. Organize services by groups to perform actions simultaneously on multiple services and to control user access to services.

Role required: evt_mgmt_admin
Service groups appear on the Event Management dashboard.

1. Navigate to Event Management > Services > Service Groups.
2. Click New.
3. In the Name field, enter the name of the new service group.
4. Optional: In the Parent Group field, enter the name of another group in which to embed the new group.
   Users with access to a parent service group automatically have access to all its child groups.
5. Click Submit.

Assign a role to a service group

Assign an Event Management role to the business service group to ensure that group members can manage and act on alerts.

Configure a service group to enable users to manage business services, application service, or alert groups.

Role required: evt_mgmt_admin

1. Navigate to Event Management > Services > Service Group Responsibilities.
2. Click New.
3. In the Business Service Group field, select the name of the service group.
4. In the Role field, select an Event Management role:
   - evt_mgmt_admin
   - evt_mgmt_integration
   - evt_mgmt_operator
   - evt_mgmt_user
5. Click Submit.
6. To find users who are assigned to the role, navigate to User Administration > Users > Roles and search for the role.

Configure Event Management domain separation

You can configure Event Management for domain separation to create logically defined domains that limit unauthorized access to data. When domains are separated in Event Management, users can only see and manage alerts and events in their own (tenant) domain.

Role required: evt_mgmt_admin and evt_mgmt_integration


The following Event Management features have limited domain separation support.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business service groups</td>
<td>Supported. Limitation: The user can define a business service that is visible in other domains with services that are not visible in other domains. In this case, other domains can see impact results but cannot see how this result was calculated.</td>
</tr>
<tr>
<td>Feature</td>
<td>Support</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Event – alert flow</td>
<td>Supported. Supported. Supported. Supported. Separation is based on the</td>
</tr>
<tr>
<td></td>
<td>domain user that sent events. User access is required for the</td>
</tr>
<tr>
<td></td>
<td>credentials of the sending API events or in the configuration of the</td>
</tr>
<tr>
<td></td>
<td>MID Server reporting events.</td>
</tr>
<tr>
<td></td>
<td>In a multi-domain environment, each MID Server can serve only one</td>
</tr>
<tr>
<td></td>
<td>domain according to the integration user that it uses. In the</td>
</tr>
<tr>
<td></td>
<td>configuration of the connector instance, make sure that the MID Server</td>
</tr>
<tr>
<td></td>
<td>uses the same domain as Event Management.</td>
</tr>
<tr>
<td>Impact calculation</td>
<td>Supported. Segregation is based on the manner in which CIs are</td>
</tr>
<tr>
<td></td>
<td>segregated.</td>
</tr>
<tr>
<td>Application services</td>
<td>Partially supported. Segregation is based on the manner in which CIs</td>
</tr>
<tr>
<td></td>
<td>are segregated.</td>
</tr>
<tr>
<td>Technical services</td>
<td>Partially supported. Segregation is based on the manner in which CIs</td>
</tr>
<tr>
<td></td>
<td>are segregated.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The discovery process does not segregate CIs by domain.</td>
</tr>
<tr>
<td>Remediation</td>
<td>Supported. While editing alert management rules, users can only apply</td>
</tr>
<tr>
<td></td>
<td>relevant workflows.</td>
</tr>
<tr>
<td></td>
<td>For information on invoking domain separation through subflows, see</td>
</tr>
<tr>
<td></td>
<td><a href="#">Invoke domain separation through subflows</a>.</td>
</tr>
<tr>
<td></td>
<td>For more information on domain separation in the Flow Designer, see</td>
</tr>
<tr>
<td></td>
<td><a href="#">Domain separation and Flow Designer</a>.</td>
</tr>
<tr>
<td></td>
<td>1. If it is not already active, activate the Domain Support – Domain</td>
</tr>
<tr>
<td></td>
<td>Extension Installer plugin.</td>
</tr>
<tr>
<td></td>
<td>2. Configure a connector instance to use a MID Server from the same</td>
</tr>
<tr>
<td></td>
<td>domain as Event Management.</td>
</tr>
</tbody>
</table>

Get top-down discovery service mapping
A top-down discovery provides a list of CIs and their interrelationships. This information is useful for managing software services and hardware issues that are associated with alerts.

Role required: evt_mgmt_admin
You can use Service Mapping to get top-down discovery information. Service Mapping, which is closely related to Event Management, provides tools to discover the relationships between network, hardware, software, storage, and other devices.

In Event Management, Service Mapping relationships appear on alerts. You can view the related CI and other information in the Event Management dashboard. The mapping relationships
appear in a business service map with the generated alert, any CIs that are bound to the alert, and all related CIs.

*Rotate event tables and alert table cleanup*

The growth of data tables impedes performance. Preserve instance performance by event table rotation and alert table cleanup for status and alert history retention.

**Containing the growth of event tables**

By default, the Event Management event (em_event) table uses table rotation. The table rotation schedule ensures that new data is written to one of the tables in the rotation group. According to the value of **Clean base rotation** setting, the base table is deleted and then a new round of writing the tables in a daily rotation commences. By default, this table deletion occurs after seven rotations.

To back up events to a custom table, see *Event Management configuration preferences*.

**Containing the growth of alert tables**

By default, a scheduled job cleans out the alert (em_alert) table. Alert table cleanup is recommended for status and alert history retention.

**Event table rotation**

Table rotation is used by Event Management, by default, to contain the growth of event (em_event) tables within the rotation table group.

**Role required:** evt_mgmt_admin

Table rotation ensures that event tables grow to a reasonable size.

Each table rotation has a schedule, according to which new data is written to one of the tables in the rotation group. A table rotation group includes the original table plus several additional tables. After the rotation procedure has written the last table in a rotation, the rotation overwrites the first table in the rotation. By default, the table rotation for Event Management is to rotate the em_table to an offload table and the base table cleanup takes place after seven table rotations.

**Purge status and alert history**

You can configure automatic table cleanup to improve performance. Automatic table cleanup is recommended for status and alert history retention.

**Role required:** evt_mgmt_admin

If records are older than 90 days, by default they are purged. If records are retained for a longer period, larger volumes of data are stored, and performance may be impacted.

**Note:** Event (em_event) table growth is contained, by default, by the use of table rotation. For information about backing up events to a custom table, see *Event Management configuration preferences*.

1. **Navigate to Automated Test Framework > Administration > Table Cleanup.**
2. **If required, shorten the retention period (in seconds) for records of the following tables:**
   - Alerts (em_alert) (Default: 7,776,000)
   - Alert History (em_alert_history) (Default: 7,776,000)
   - Impact Status (em_impact_status) (Default: 7,776,000)

**Alert table clean-out**
The Schedule Jobs feature executes a script that closes alerts, in the alerts (em_alert) table, that meets specified conditions.

Role required: evt_mgmt_admin

When the Event Management plugin is activated, by default a job is scheduled to run the Event Management - auto close alerts script to automatically close alerts in the alerts table that meet the specified conditions.

The Event Management - auto close alerts script runs automatically and repeatedly at the scheduled time and closes alerts that meet the specified conditions.

Configure email notification on business service severity change
Configure an email notification to notify users when there is a business service severity change.

Role required: evt_mgmt_admin

Configure an email notification to notify users when there is a business service severity change. The notification involves specifying when to send it, who receives it, and what it contains.

1. Navigate to System Properties > Email Properties.
2. In the Outbound Email Configuration section, in the Email sending enabled option, select Yes to enable sending severity information by email.
3. Click Save.
4. Navigate to System Notification > Email > Notifications.
5. Click New.
   a) In the Email Notification Actions form, specify these values.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Type a descriptive name, for example, Business Service severity change.</td>
</tr>
<tr>
<td>Table</td>
<td>Select the required service type and its corresponding table: Alert Group, Discovered Service, Technical Service, or Application service.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to enable email notification.</td>
</tr>
</tbody>
</table>

b) Click the Who will receive tab. In the Users area, search for and select the required user (the user does not require System Administrator credentials). Repeat this step for as many users that are required to receive this notification.

c) Click the When to send tab. In the choose field box, search for and select Severity.

d) Click the What will it contain tab. In the Subject field, type meaningful text that makes it clear to the receiver of the email notification what the content of the email is.

e) In the Message HTML message area, using the Select variables functions, compose the conditions under which the email notification is to be sent. For example, configure these variables to change severity:

- severity changed
- Severity: ${severity}
- on business service
- Name: ${name}
6. Click Submit.

View Event Management license usage
Event Management is licensed based on the number of nodes that can send events to the instance directly or through a third-party monitoring tool during the last year.

Role required: evt_mgmt_admin or evt_mgmt_operator

The Event Management - Node Count job calculates the number of unique nodes that send event information, either directly or through third-parties, to Event Management and are eligible for licensing in the last year. The information from this job is stored in the License Usage (em_unique_nodes) table for visibility by all users. All nodes that send event information are counted, whether the alert generated from the event was bound to a CI or not.

Navigate to Event Management > Settings > License Usage. The License Usage list appears.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDB CI</td>
<td>The CI that is bound to the alert that is generated from the event. If there is a value for this field, the Node field is empty.</td>
</tr>
<tr>
<td>Node</td>
<td>The string value of the Node field of the event. If there is a value for this field, the CMDB CI field is empty.</td>
</tr>
<tr>
<td>Type</td>
<td>One of these values:</td>
</tr>
<tr>
<td></td>
<td>• Unknown - A CI has not bound to the generated alert.</td>
</tr>
<tr>
<td></td>
<td>• Server - The source of the event was a server.</td>
</tr>
<tr>
<td></td>
<td>• PaaS - The source of the event was Platform as a Service.</td>
</tr>
<tr>
<td>Is licensable</td>
<td>One of these values:</td>
</tr>
<tr>
<td></td>
<td>• True - The event listed is counted towards the license usage.</td>
</tr>
<tr>
<td></td>
<td>• False - The event listed is not counted towards the license usage.</td>
</tr>
</tbody>
</table>

Configure contextual colors and icons
Use the Contextual colors and icons form to configure color, text, and icons to have different default or custom contexts, identified by a context ID.

Role required: evt_mgmt_admin

You can customize the colors or icons of elements in the Contextual colors and icons form to affect the displayed colors or icons of severities in related Event Management screens, either when accessibility mode is off or on. For more information about accessibility, see Accessibility features.

In each color selection field, you have the option of entering the value of the color as any of the following Cascading Style Sheet (CSS) formats that determine how HTML elements should be displayed:

- Name: predefined color names, for example, red, green, olive, blue, darkblue, or lightblue
- RGB decimal: RGB(102, 153, 204)
- RGB hex: #223344

See the following websites for information about CSS color declarations (including hex or RGB notation):
- HTML Colors (W3CSchools) for more information on hex and RGB notation.
- HTML Color Names (W3CSchools) for valid color names.

The context IDs that are provided with the base instance are:
- information_severity
- warning_severity
- minor_severity
- major_severity
- critical_severity
- topology_connection

1. Navigate to Event Management > Settings > Contextual colors and icons.
2. Click New or modify an existing configuration.
3. Configure the following fields.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context details</td>
<td></td>
</tr>
<tr>
<td>Context ID</td>
<td>Enter a unique ID for the context, for example,</td>
</tr>
<tr>
<td></td>
<td>clear_severity.</td>
</tr>
<tr>
<td>Context Name</td>
<td>Readable name for the context. This text is</td>
</tr>
<tr>
<td></td>
<td>available to be read out by screen readers.</td>
</tr>
<tr>
<td>Batch</td>
<td>The batch that this context is a member of.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of uses for the context.</td>
</tr>
<tr>
<td>Colors</td>
<td></td>
</tr>
<tr>
<td>CSS Color</td>
<td>The corresponding CSS color code for the context</td>
</tr>
<tr>
<td></td>
<td>color name.</td>
</tr>
<tr>
<td>CSS Color on Interaction</td>
<td>CSS color code that appears on hover or focus.</td>
</tr>
<tr>
<td>Accessible CSS Color</td>
<td>Accessible CSS color code that appears when</td>
</tr>
<tr>
<td></td>
<td>accessibility mode is enabled.</td>
</tr>
<tr>
<td>Accessible CSS Color on Interaction</td>
<td>Accessible CSS color code that appears on hover or focus.</td>
</tr>
<tr>
<td>Icons</td>
<td></td>
</tr>
<tr>
<td>Icon</td>
<td>Icon that is displayed when the context includes</td>
</tr>
<tr>
<td></td>
<td>an icon.</td>
</tr>
<tr>
<td>Accessible Icon</td>
<td>Icon that is displayed when the context includes</td>
</tr>
<tr>
<td></td>
<td>an icon and when accessibility mode is enabled.</td>
</tr>
<tr>
<td></td>
<td>To add your own icon, or to see a list of</td>
</tr>
<tr>
<td></td>
<td>available icons, see Upload one or more images.</td>
</tr>
</tbody>
</table>

4. Click Submit.
Administer events

An event is a notification from one or more monitoring tools that indicate something of interest has occurred, such as a log message, warning, or error.

The manner in which events process and alerts generate are based on the following factors:

- Event collection configuration via MID Server, script, SNMP trap collector, email.
- Event rule, alert binding, and event field mapping configuration for processing events from various sources.

You can view event information in the following places:

- The event list view displays the entire list of events.
- Under Event rules, see Create or edit an event rule, you can view a list of individual events or a group of events that are not associated with any event rule.
- If an event rule or event field mapping used the event or group of events to generate an alert, the event information appears in the Activity section of the alert.

View events

Event Management tracks individual events to manage external systems. An event is a notification from one or more monitoring tools that indicate that something of interest has occurred, such as a log message, warning, or error. Event Management receives or pulls events from one or more external event sources and stores them in the Event (em_event) table. Event Management provides a list of raw incoming events.

Role required: evt_mgmt_admin, evt_mgmt_operator, evt_mgmt_user, or evt_mgmt_integration

Note: Business rules must not change the Category field on event (em_event) tables.

The event monitoring tool generates the values of the source and resource fields. Event Management implementers can define event types and register nodes to help uniquely identify incoming events and create alerts for the specific needs of the enterprise. Event Management uses this information to determine whether to create an alert or update an existing one.

An event source may generate duplicate events with the same identifying information. For events with the same identifying information, Event Management uses the time interval between events to determine if events represent an existing issue or new issue.

Additional fields should be included in the Additional information field of the event. Do not add additional fields to an event by adding a custom field to the event (em_event) table. For more information about how to include additional fields in events, see Custom alert fields.

Note: Avoid writing business rules for event (em_event) tables as they can result in performance degradation.

1. Navigate to Event Management > All Events.

The All Events list displays the following columns.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>Populated by</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time of event</td>
<td>The time that the event occurred, in the network node time zone.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
<td>Populated by</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Description</td>
<td>Reason for event generation. Shows extra details about an issue. For example, a server stack trace or details from a monitoring tool. This field has a maximum length of 4000 digits.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Node</td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100 digits.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Type</td>
<td>Pre-defined event type, such as high CPU, which is used to identify an event record. This field has a maximum length of 100 digits.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Resource</td>
<td>Node resource that is relevant to the event. For example, Disk C, CPU-1, the name of a process, or service. This field has a maximum length of 100 digits.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Message Key</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the <strong>Source</strong>, <strong>Node</strong>, <strong>Type</strong>, <strong>Resource</strong>, and <strong>Metric Name</strong> field values. This field has a maximum length of 1024 digits.</td>
<td>External event monitoring tool</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
<td>Populated by</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>State</td>
<td>The status of the event:</td>
<td>Event</td>
</tr>
<tr>
<td></td>
<td>• <strong>Ready</strong>: Event has been received and is waiting to be processed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Queued</strong>: Event is queued by the event processor job.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Processed</strong>: Event was successfully processed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Error</strong>: Failure occurred while processing the event. For example, the</td>
<td></td>
</tr>
<tr>
<td></td>
<td>event collection method or event <strong>Severity</strong> is blank.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Ignored</strong>: Value is not in use.</td>
<td></td>
</tr>
<tr>
<td>Severity</td>
<td>Mandatory. The options are typically interpreted as follows:</td>
<td>Event</td>
</tr>
<tr>
<td></td>
<td>• <strong>Critical</strong>: Immediate action is required. The resource is either not</td>
<td></td>
</tr>
<tr>
<td></td>
<td>functional or critical problems are imminent.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Major</strong>: Major functionality is severely impaired or performance has</td>
<td></td>
</tr>
<tr>
<td></td>
<td>degraded.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Minor</strong>: Partial, non-critical loss of functionality or performance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>degradation occurred.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Warning</strong>: Attention is required, even though the resource is still</td>
<td></td>
</tr>
<tr>
<td></td>
<td>functional.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Info</strong>: An alert is created. The resource is still functional.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>Clear</strong>: No action is required. An alert is not created from this event.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Existing alerts are closed.</td>
<td></td>
</tr>
<tr>
<td>Alert</td>
<td>If an alert was created as a result of the event, this field contains the</td>
<td>A matching event rule or event field</td>
</tr>
</tbody>
</table>
For further ways to see which event rules were applied to the event, in the Events list, you can optionally select to display the Event rule and Processing Notes fields.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching Event rule</td>
<td>Contains the event rule that matches the current event rule definitions. This information is primarily used to create event rule recommendations for events that currently are not covered by rules. If event rules have been modified, this information can be retrieved from the Processing Notes field, described below.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This field does not show the history of event rules that were applied to the event.</td>
</tr>
<tr>
<td>Processing Notes</td>
<td>View information related to the binding to CI type and the event rules that were applied to the event.</td>
</tr>
</tbody>
</table>

4. To create an event rule if an event is not associated with an event rule:
   a) Click the required event. The event opens in the Event form.
   b) Click Create Event Rule.
      The fields of the new Event rule are populated with data from the event. You can use the event information, for example, when composing fields for alerts.

**Event identifiers**

Event fields uniquely identify each event. Event Management uses this information to determine whether to create a new alert or update an existing one.
By default, each event is uniquely identified by the **Message Key**. If the **Message Key** is not populated, a concatenation of the **Source**, **Type**, **Node**, **Resource**, and **Metric Name** fields are used and these fields populate the **Message Key**. If identifiers are not supplied in the event, you can add them with event rules.

**Automatic CI creation**

IT and Operational Intelligence events pushed from specified sources can automatically create CIs.

The source being configured must be registered in the system.

Role required: evt_mgmt_admin

You can configure from which source Event Management automatically creates CIs in cases where CIs could not be found in the CMDB. For the registration of CIs that result from raw Operational Intelligence data, see [Create or configure data source registration](#).

1. In the application navigation filter, enter `sa_metric_registration.list`. Existing registrations are displayed.

2. Configure an existing registration or click **New**.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Specify the name of the source exactly as it appears in the <strong>Source</strong> field of the event.</td>
</tr>
<tr>
<td>Registration Mode</td>
<td>Not relevant to the creation of CIs that do not result from raw Operational Intelligence data. Specify <strong>None</strong> (default). For CIs that do result from raw Operational Intelligence data, see <a href="#">Create or configure data source registration</a>.</td>
</tr>
<tr>
<td>Type default mode</td>
<td>Not relevant to the creation of CIs that do not result from raw Operational Intelligence data. Specify <strong>None</strong> (default). For CIs that do result from raw Operational Intelligence data, see <a href="#">Create or configure data source registration</a>.</td>
</tr>
<tr>
<td>Generate Missing CIs</td>
<td>Select to create CIs automatically.</td>
</tr>
</tbody>
</table>

3. Click **Submit**.

**Event field format for event collection**

Event Management requires all events to use a standard form, regardless of how they arrive at the instance.

In the application navigation filter, enter `em_event.list`.

**Event form**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>em_event.source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>(Source)</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>em_event.node (Node)</td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>em_event.type (Type)</td>
<td>Optional. Pre-defined event type, such as high CPU, which is used to identify an event record. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>em_event.resource (Resource)</td>
<td>Node resource that is relevant to the event. For example, Disk C, CPU-1, the name of a process, or service. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>metric_name (Metric Name)</td>
<td>Unique name to identify the metric.</td>
</tr>
<tr>
<td>em_event.event_class (Source instance)</td>
<td>If the em_event.node field is not specified, it is mandatory for alerts to be created automatically. Values for the em_event.event_class field originate from either the source generating the events or by event rule. Name of the machine or software that generated the event. For example, SolarWinds on 10.22.33.44. Corresponding field display name is Source Instance.</td>
</tr>
<tr>
<td>em_event.message_key (Message key)</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the Source, Node, Type, Resource, and Metric Name field values. This field has a maximum length of 1024 digits.</td>
</tr>
<tr>
<td>em_event.ci_type</td>
<td>JSON string that represents a configuration item. For example, {&quot;name&quot;:&quot;SAP ORA01&quot;,&quot;type&quot;:&quot;Oracle&quot;}. The CI identifier that generated the event appears in the Additional information field. This field has a maximum length of 1000 digits. Note: Reference pop-ups and click-throughs are hidden by default for read-only fields. For Configuration item and other read only fields, you can optionally change the read-only setting. For more information, see Configure pop-ups on read-only fields.</td>
</tr>
</tbody>
</table>
| em_event.severity (Severity) | Event severity options are:  
  - **Critical**: Immediate action is required. The resource is either not functional or critical problems are imminent.  
  - **Major**: Major functionality is severely impaired or performance has degraded.  
  - **Minor**: Partial, non-critical loss of functionality or performance degradation occurred.  
  - **Warning**: Attention is required, even though the resource is still functional.  
  - **Info**: An alert is created. The resource is still functional.  
  - **Clear**: No action is required. An alert is not created from this event. Existing alerts are closed. |
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>em_event.resolution_state</strong></td>
<td>Optional. If the field is empty, the resolution on corresponding alerts is still pending. Event state from the event source is either <strong>New</strong> or <strong>Closing</strong>.</td>
</tr>
<tr>
<td>(Resolution state)</td>
<td>- <strong>New</strong>: the resolution on corresponding alerts is open.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Closing</strong>: event state closes corresponding alerts.</td>
</tr>
<tr>
<td><strong>em_event.time_of_event</strong></td>
<td>Time that the event occurred in the source system. This field is a GlideDateTime field in UTC or GMT format. This field has a maximum length of 40 digits.</td>
</tr>
<tr>
<td>(Time of event)</td>
<td></td>
</tr>
<tr>
<td><strong>em_event.state</strong></td>
<td>Current processing state of the event:</td>
</tr>
<tr>
<td>(State)</td>
<td>- <strong>Ready</strong>: Event has been received and is waiting to be processed.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Processed</strong>: Event was successfully processed.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Ignored</strong>: Value is not in use.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error</strong>: Failure occurred while processing the event. For example, the event collection method or event Severity is blank.</td>
</tr>
<tr>
<td><strong>em_event.alert</strong></td>
<td>If an alert was created as a result of the event, this field contains the unique ID that Event Management generates to identify the alert.</td>
</tr>
<tr>
<td>(Alert)</td>
<td></td>
</tr>
<tr>
<td><strong>em_event.description</strong></td>
<td>Reason for event generation. Shows extra details about an issue. For example, a server stack trace or details from a monitoring tool. This field has a maximum length of 4000 digits.</td>
</tr>
<tr>
<td>(Description)</td>
<td></td>
</tr>
<tr>
<td><strong>em_event.additional_info</strong></td>
<td>Optional. A JSON string that gives more information about the event. The JSON data is supported for String values only, other value types are not supported. You must convert numbers to String values by enclosing them in double quotes. For example, this value is not supported:</td>
</tr>
<tr>
<td>(Additional information)</td>
<td><code>{&quot;CPU&quot;:100}</code> while this value is supported: <code>{&quot;CPU&quot;:&quot;100&quot;}</code>, Another example of a valid JSON string is: <code>{evtComponent&quot;:&quot;Microsoft-Windows-WindowsUpdateClient&quot;,&quot;evtMessage&quot;:&quot;Installation Failure: Windows failed. Error 0x80070490\&quot;}</code>.</td>
</tr>
<tr>
<td></td>
<td>This information can be used for third-party integration or other post-alert processing. Values in the <strong>Additional information</strong> field of an Event that are not in JSON key/value format are normalized to JSON key/value format when the event is processed. For example, assume that the following plain text is in the Additional information field &quot;Connection instance is successful&quot;.</td>
</tr>
<tr>
<td></td>
<td>When the event is processed, all this plain text becomes one JSON string and might not be useful within an alert. In the resultant alert, this string is in the <strong>Additional information</strong> field in JSON key/value format, containing the data:</td>
</tr>
<tr>
<td></td>
<td><code>{&quot;additional_content&quot;: &quot;Connection instance is successful&quot;}</code>.</td>
</tr>
</tbody>
</table>
### Event field mapping configuration

Use Event field mappings rules to map values from specific fields to values in other fields.

Event Management stores event field mappings in the Event Field Mapping (em_mapping_rule) table. The mappings apply after event rule processing and prior to alert generation. The mapping values from the Event Mapping Pair (em_mapping_pair) table apply to the alert. The original event severity remains unchanged.

For example, if events come with the field `org_severity` with the values `Low, Medium, High` and you want the alert Severity to hold this value, create an event field mapping rule that maps the field `org_severity` to Severity, with these values:

**Example showing org_severity mapped to Severity**

<table>
<thead>
<tr>
<th>Original org_severity value</th>
<th>Map to alert Severity value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Warning</td>
</tr>
<tr>
<td>Medium</td>
<td>Major</td>
</tr>
<tr>
<td>High</td>
<td>Critical</td>
</tr>
</tbody>
</table>

### Default event field mappings

Event Management provides default event field mappings for commonly used system monitoring tools. The Event Mapping Pairs from event field mappings format the incoming event data for Event Management.

You can view the default event field mappings and mapping pairs by navigating to Event Management > Rules > Event Field Mappings and double-clicking Name. The default event field mappings that are available for the following event sources:

- enterprises.20006.1.5
- enterprises.20006.1.7
- HPOMWIN
- Hyperic
- Microsoft Operations Manager (mom)
- netappDataFabricManager
- Oracle EM
- oraEM4Traps
- SNMPv2 Generic Trap
- SolarWinds
- Trap From Enterprise 9
- vCenter
- vmwVC
- whatsup-whatsupState
For HP Operations Manager, the following default event field mappings are active by default in the **HPOMWIN** Source category.

**Default HP Operations Manager event field mappings**

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPOMWIN-hpomSeverity</td>
<td>Maps the source <code>hpomSeverity</code> field to the event severity.</td>
</tr>
</tbody>
</table>

**NetApp DataFabric Manager**

For NetApp DataFabric Manager, the following default event field mappings are active by default in the **netappDataFabricManager** Source category.

**Default NetApp DataFabric Manager event field mappings**

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>netappDataFabricManager-dfmEventSeverity</td>
<td>Maps the source <code>dfmEventSeverity</code> field to the event severity.</td>
</tr>
</tbody>
</table>

**NetDiscover**

For NetDiscover, the following default event field mappings are active by default in the **enterprises.20006.1.5** and **enterprises.20006.1.7** Source categories.

**Default Net Discover event field mappings**

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enterprises.20006.1.5-trap_severity</td>
<td>Maps the source <code>trap_severity</code> field to the event severity.</td>
</tr>
<tr>
<td>enterprises.20006.1.7-trap_severity</td>
<td>Maps the source <code>trap_severity</code> field to the event severity.</td>
</tr>
</tbody>
</table>

**Microsoft Operations Manager**

For Microsoft Operations Manager, the following default event field mappings are active by default in the **mom** Source category.

**Default Microsoft Operations Manager event field mappings**

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mom-momSeverity</td>
<td>Maps the source <code>momSeverity</code> field to the event severity.</td>
</tr>
</tbody>
</table>

**Oracle**

For Oracle, the following default event field mappings are active by default in the **oraEM4Traps** Source category.
Default Oracle event field mappings

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oraEM4Traps-oraEM4AlertSeverity</td>
<td>Maps the source oraEM4AlertSeverity field to the event severity.</td>
</tr>
</tbody>
</table>

SNMP traps

For SNMP Traps, the following default event field mappings are active by default in the SNMPv2 Generic Trap Source category.

Default SNMP Traps event field mappings

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ifOperStatus</td>
<td>Maps the source ifOperStatus field to the ifOperStatus.</td>
</tr>
<tr>
<td>ifAdminStatus</td>
<td>Maps the source ifAdminStatus field to the ifAdminStatus.</td>
</tr>
</tbody>
</table>

SolarWinds

For SolarWinds, the following default event field mappings are active by default in the SolarWinds Source category.

Default SolarWinds event field mappings

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>solarwinds-icon-severity</td>
<td>Maps the source icon field to the event severity.</td>
</tr>
<tr>
<td>solarwinds-Availability-severity</td>
<td>Maps the source Availability field to the event severity.</td>
</tr>
<tr>
<td>solarwinds-ComponentStatus-severity</td>
<td>Maps the source ComponentStatus field to the event severity.</td>
</tr>
<tr>
<td>solarwinds-Status-severity</td>
<td>Maps the source Status field to the event severity.</td>
</tr>
<tr>
<td>solarwinds-type</td>
<td>Maps the source eventType field to the event type.</td>
</tr>
</tbody>
</table>

Trap from Enterprise 9

For Trap from Enterprise 9, the following default event field mappings are active by default in various Source categories.

Default Trap from Enterprise 9 event field mappings

<table>
<thead>
<tr>
<th>Event field mapping name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mnState (mnState category)</td>
<td>Maps the source mnState field to the mnState.</td>
</tr>
<tr>
<td>tslineSesType (tslineSesType category)</td>
<td>Maps the source tslineSesType field to the tslineSesType.</td>
</tr>
<tr>
<td>cisco.snmpTrapOID (snmpTrapOID category)</td>
<td>Maps the source snmpTrapOID field to the snmpTrapOID.</td>
</tr>
</tbody>
</table>
### Event field mapping name | Description
--- | ---
ciscoFlashCopyStatus (ciscoFlashCopyStatus category) | Maps the source `ciscoFlashCopyStatus` field to the `ciscoFlashCopyStatus`.
tcpConnState (tcpConnState category) | Maps the source `tcpConnState` field to the `tcpConnState`.

### VMware vCenter

For VMware VCenter, the following default event field mappings are active by default in the `vmwVpxdNewStatus` Source category.

**Default VMware VCenter event field mappings**

| Event field mapping name | Description |
--- | ---
vmwVC-vmwVpxdNewStatus | Maps the source `vmwVpxdNewStatus` field to the event `severity`.

### Whatsup

For *whatsup*, the following default event field mappings are active by default in the *whatsup* Source category.

**Default whatsup event field mappings**

| Event field mapping name | Description |
--- | ---
whatsup-whatsupState | Maps the source `whatsupState` field to the event `severity`.

### Create event field mappings

Use event field mappings to provide more comprehensive information in an event alert by substituting values from the event field mapping rule into the event.

Role required: `evt_mgmt_admin`

Create the rule to match the event by its class and original values. Also specify the new values to replace the original values in the event.

1. Navigate to **Event Management > Rules > Event Field Mapping**.
2. Click **New** or open an existing rule to edit.
3. Fill in the fields, as appropriate.
### Event Field Mapping form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Event field mapping name.</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Order</td>
<td>Number to define the order in which this action should be processed. Actions with lower numbers are processed first.</td>
</tr>
</tbody>
</table>
Mapping type

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping type</td>
<td>Mapping mechanism that is used to change an event field value.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Single field</strong>: Mapping rule that transforms specific values from one event field to another event field. For example, whenever the <code>ciscoFlashCopyStatus</code> mapping rule finds the specific value 8 in the <code>ciscoFlashCopyStatus</code> name-value pair, the mapping rule updates the field value to <code>copyDeviceBusy</code>.</td>
</tr>
<tr>
<td></td>
<td>· <strong>Constant</strong>: Mapping rule that transforms any value in the specified field to the new value provided. For example, a mapping rule could transform any value in the <code>Node</code> field to a hard-coded value such as <code>Linux1</code>.</td>
</tr>
</tbody>
</table>

Active

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Check box that activates or deactivates the event field mapping. If possible, find and apply another event field mapping rule.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select **Save**.

5. If you selected **Single field**, fill in the fields, as appropriate.

**Single value fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>From field</td>
<td>Event field to replace.</td>
</tr>
<tr>
<td>To field</td>
<td>Event field where the mapping rule inserts or updates the value. When this field is identical to the <code>From field</code>, the mapping rule updates the value in memory of the event field.</td>
</tr>
</tbody>
</table>

6. If you selected **Constant**, fill in the fields, as appropriate.

**Constant fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Value you want to use for the <code>To field</code>. This field appears when the Mapping type is <code>Constant</code>.</td>
</tr>
<tr>
<td>Value (Event Mapping Pairs section)</td>
<td>Value you want to insert or update into the <code>To field</code>. The mapping rule overwrites any existing value in the <code>To field</code>. This field appears when the Mapping type is <code>Single field</code>.</td>
</tr>
</tbody>
</table>

7. In the **Key** field, fill in the fields, as appropriate. Click + to add more **Key** fields, as required.
Key fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key (Event Mapping Pairs section)</td>
<td>Value that the mapping rule searches for. Whenever the event field has this value, the mapping rule adds the value listed in the Value field to the field listed in the To field. This field appears when the Mapping type is Single field.</td>
</tr>
</tbody>
</table>

8. Click **Submit**.

For example, see these values for a predefined rule that is applied to events in the Trap From Enterprise 9 class. If the events contain the `snmpTrapOID` element with a value of `iso.org.dod.internet.private.enterprises.cisco.0.0`, the mapping rule changes the value to `reload` in alerts. If the events contain the `snmpTrapOID` element, a value of `iso.org.dod.internet.private.enterprises.cisco.0.1`, the mapping rule changes the value to `tcpConnectionClose` in alerts.

<table>
<thead>
<tr>
<th>Field</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>cisco.snmpTrapOID</td>
</tr>
<tr>
<td>Source</td>
<td>Trap From Enterprise 9</td>
</tr>
<tr>
<td>Mapping type</td>
<td>Single field</td>
</tr>
<tr>
<td>From field</td>
<td>snmpTrapOID</td>
</tr>
<tr>
<td>To field</td>
<td>snmpTrapOID</td>
</tr>
<tr>
<td>Event Mapping Pairs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Pair 1</td>
</tr>
<tr>
<td></td>
<td>· Key: <code>iso.org.dod.internet.private.enterprises.cisco.0.0</code></td>
</tr>
<tr>
<td></td>
<td>· Value: <code>reload</code></td>
</tr>
<tr>
<td></td>
<td>· Pair 2</td>
</tr>
<tr>
<td></td>
<td>· Key: <code>iso.org.dod.internet.private.enterprises.cisco.0.1</code></td>
</tr>
<tr>
<td></td>
<td>· Value: <code>tcpConnectionClose</code></td>
</tr>
</tbody>
</table>

Test an event field mapping by sending an event that contains a field that is present in the event field mapping.

**Manually create and send events**

You can manually create and send events to confirm that Event Management properly manages events and generates alerts.

Role required: evt_mgmt_admin

For example, you can manually send events to:

- Confirm that the MID Server is using an event connector definition and instance to send events.
- Confirm that event rules, event field mapping, and other configurations do generate alerts.
• Track an operation or action that did not generate an event.

1. Navigate to Event Management > All Events.
2. Click New.
3. On the form, fill in the fields.

**Event form**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Node</td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Type</td>
<td>Pre-defined event type, such as high CPU, which is used to identify an event record. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Resource</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the Source, Node, Type, Resource, and Metric Name field values. This field has a maximum length of 1024 digits.</td>
</tr>
<tr>
<td>Metric Name</td>
<td>Unique name to identify the metric.</td>
</tr>
<tr>
<td>Source instance</td>
<td>Name of the machine or software that generated the event. For example, SolarWinds on 10.22.33.44. Corresponding field display name is Source Instance.</td>
</tr>
<tr>
<td>Message key</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the Source, Node, Type, Resource, and Metric Name field values. This field has a maximum length of 1024 digits.</td>
</tr>
<tr>
<td>Severity</td>
<td>The severity of the event. The options are typically interpreted as follows:</td>
</tr>
<tr>
<td></td>
<td>• Critical: Immediate action is required. The resource is either not functional or critical problems are imminent.</td>
</tr>
<tr>
<td></td>
<td>• Major: Major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td></td>
<td>• Minor: Partial, non-critical loss of functionality or performance degradation occurred.</td>
</tr>
<tr>
<td></td>
<td>• Warning: Attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>• Clear: No action is required. An alert is not created from this event. Existing alerts are closed.</td>
</tr>
<tr>
<td></td>
<td>• Info: An alert is created. The resource is still functional.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Resolution state        | Optional. If the field is empty, the resolution on corresponding alerts is still pending. Event state from the event source is either **New** or **Closing**.  
  - **New**: the resolution on corresponding alerts is open.  
  - **Closing**: event state closes corresponding alerts. |
| Time of event           | Time that the event occurred in the source system. This field is a GlideDateTime field in UTC or GMT format. This field has a maximum length of 40 digits. |
| State                   | The current processing state of the event:  
  - **Ready**: Event has been received and is waiting to be processed.  
  - **Processed**: Event was successfully processed.  
  - **Ignored**: Value is not in use.  
  - **Error**: Failure occurred while processing the event. For example, the event collection method or event **Severity** is blank. |
| Alert                   | If an alert was created as a result of the event, this field contains the unique ID that Event Management generates to identify the alert. |
| Description             | Reason for event generation. Shows extra details about an issue. For example, a server stack trace or details from a monitoring tool. This field has a maximum length of 4000 digits. |
### Additional information

A JSON string that gives more information about the event. The JSON data is supported for String values only, other value types are not supported. You must convert numbers to String values by enclosing them in double quotes. For example, this value is not supported: `{"CPU":100 }` while this value is supported: `{"CPU":"100"}`. Another example of a valid JSON string is: `{"evtComponent":"Microsoft-Windows-WindowsUpdateClient","evtMessage":"Installation Failure: Windows failed. Error 0x80070490"}`. This information can be used for third-party integration or other post-alert processing. Values in the **Additional information** field of an Event that are not in JSON key/value format are normalized to JSON key/value format when the event is processed. For example, assume that the following plain text is in the **Additional information** field "Connection instance is successful". When the event is processed, all this plain text becomes one JSON string and might not be useful within an alert. In the resultant alert, this string is in the **Additional information** field in JSON key/value format, containing the data: {"additional_content": "Connection instance is successful"}.

4. Click **Submit**.

### Event rules

Use event rules to generate alerts for tracking and remediation. Event rules are stored in the Event Rule (em_match_rule) table. Configure and customize event rules to manage events and alert generation.

Event rules do not change the event records in the Event table. Changes to event data are stored in the ServiceNow instance memory.

Use the Event rule designer to create and configure event rules. See the following video:

You can use the default event rules or event rules that you have created to:

- Apply an event rule filter to determine whether the rule applies to an event.
- Apply a transform with **Event Match Fields** and optional **Event Compose Fields** to format the alert text.
- Apply a threshold to create custom alerts for rapidly recurring events.
- Automatically create or close alerts.
- Bind alerts to CI information from the CMDB.
Create or edit an event rule
You can create event rules to generate alerts for tracking and remediation.

Role required: evt_mgmt_admin

You can view the list of available event rules on the event rule page.

You can create rules that:

- Transform information in events to populate specified alert field values and compose alert fields from various values.
- Configure threshold rules that create or close alerts only when the incoming matching events exceed the specified threshold.
- Bind alerts to CIs using CI identifiers.

Options to create the rule are:

- Create an event rule and assign event fields for alert generation.
- Create a rule from an existing event or groups of events that do not have a rule. In this case, the event fields are copied to the Event Match Fields section of the rule.
- Edit an existing event rule.

Note: Event rules that are not configured to perform any action are skipped. Therefore, if the rule is not configured as ignore, threshold, or binding, it is important to specify either the match or the compose fields.

1. Navigate to Event Management > Rules > Event Rules and take one of the following actions:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create an event rule from an existing event</td>
<td>1. Click the link for unassociated events or grouped events that are not mapped to rules.</td>
</tr>
<tr>
<td></td>
<td>2. Select the event that you want to use for creating the rule.</td>
</tr>
<tr>
<td></td>
<td>The event fields are copied to the Event Field Rules section of the rule.</td>
</tr>
</tbody>
</table>
## Edit an existing event rule

In the event rule list, click the required event rule to be modified. The event rule opens in the event rule designer where you can modify the values of the fields.

- Click **Save and Upgrade**

## Create an event rule

Click **New**.

### 2.

Ensure that **Active** is selected. When the rule is deactivated, Event Management finds and applies another event rule. An alert is still created for the event unless **Ignore** is selected in another applicable rule or when configuring the filter for this event rule.

### 3.

Enter a unique and meaningful name and fill in the form.

### Event Rule Info form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Category to which this matching rule applies. The mapping rule only applies to events with the same event class value. If this value is empty, apply the rule to all events.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which an event rule is evaluated when multiple rules are defined for the same type of event. Event rules are evaluated in ascending order.</td>
</tr>
<tr>
<td>Description</td>
<td>Type additional information that describes the event rule.</td>
</tr>
</tbody>
</table>

### 4.

Optional: Define the event rule using these Event Rule designer features:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Event Filter</strong></td>
<td>Define a filter to restrict to which events the event rule must apply. See <a href="#">Filter the events that an event rule applies to</a>.</td>
</tr>
<tr>
<td><strong>Transform and compose alert output</strong></td>
<td>Configure the customization of alert content. See <a href="#">Configure an event rule to customize alert content</a>.</td>
</tr>
<tr>
<td><strong>Threshold</strong></td>
<td>Create or close alerts according to the specified threshold. See <a href="#">Set a threshold to suppress alert generation</a>.</td>
</tr>
</tbody>
</table>
5. Click Save, Submit, or Update.

Use event input information
The Event Input pane that is included in the steps to create an event rule provides a reference to the information that you can use when configuring an event rule.

In the Transform and compose alert output pane, you can also drag the information into the required fields to customize alert content. The fields in the right-hand pane appear as a reference to which fields, expressions, and their values are available when working in the Event Rule designer. The fields that display in the Transform and compose alert output left-hand workarea become the fields in the generated alert form.

General components and navigation
The Event Input pane consists of these sections:

- Event Additional Info
- Expressions
- Event Raw Info

The Event Input pane displays in all Event Rule designer sections, except for the Event Rule Info section. The fields in the right-pane can be modified only in the Transform and compose alert output section. In all the other sections, the Event Input pane is read-only and available for reference purposes.

Note: If Accessibility enabled is selected in System Settings, standard edit mode is not available.

Edit modes

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard edit mode</td>
<td>Text selected in the left-hand workarea cannot be edited. Undo is not available. In this mode, you can only mark the text or values and transform it by selecting a field name.</td>
</tr>
<tr>
<td>Mode</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Regex edit mode</td>
<td>Edit text in the left-hand workarea by selecting it, as required. After you have edited the text, you cannot go back to the standard mode. <strong>Undo</strong> is not available. However, <strong>Reset</strong> is available, returning the field to its original value. You can compose text using regular expression (regex) format conventions, for example, type (.*) to create a field name. You cannot use field values like: type &quot;$&quot; to include the value of a field and specify the name of the field in parentheses (). You can include special characters in a field by ‘escaping’ them, that is, by prefixing the character with a ‘\‘. For example, to include a ‘\‘ in your text, enter &quot;.</td>
</tr>
</tbody>
</table>

**Event additional info**

The Event additional info section displays default event form fields, plus fields that were created to display in alerts and that are added to the Additional information section of the alert.

When configuring text in the fields in the left-hand workarea, you can drag field names and/or their value from this section into the fields.

**Expressions**

The Expressions section is available only if an additional information field was created. You can create expressions from event raw data or from additional information data.

1. Click in a field in the Event Input pane to perform transformation actions.

2. Create an expression by transforming information and associating it with a field name. Either select a field name from the suggested list, or create a field name, as required.

<table>
<thead>
<tr>
<th>EXPRESSIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric Name</td>
</tr>
<tr>
<td>High condition</td>
</tr>
</tbody>
</table>

3. Click in a field in the Event Input pane. In the left-hand workarea, select the required information.

Create expressions by transforming information and associating it with a field name. Select a field name from the suggested list, or you can create a field name, as required.

**Event raw info**

The Event raw info section is always available. Its contents are the fields from the Event form.

To transform text, click a field in the right-panel to open it in the work area on the left-panel, in the Mark Expressions section. You can select text in the Mark Expressions section where the selected
text becomes the value of a newly created field. Name the new field. This new field is added to the Expressions section.

**Note:** When creating regex expression for input, ensure that you create input with the variables in sequential order. While variables can be skipped for inclusion, the selected order must be maintained.

Controlling how fields appear:

- Use the blacklist (em_event_rule_black_list) table to hide fields from the right-pane.
- Use the em_event_rule_order form to order the position of each field.
- Fields that are not in the blacklist and also not part of the em_event_rule_order table, still appear in the event rule designer, in the right-panel and in the **Transform and Compose Alert Output** tab.

**Filter the events that an event rule applies to**

Define a filter to restrict to which events the event rule must apply. Configure the filter by providing a set of conditions that each event must match to be either excluded or included from applying to the event rule.

Ensure that a name is specified for the event rule.

Role required: evt_mgmt_admin

To construct the required condition statement, you can use event fields as well as additional fields that are defined in the **Additional Information** field of the event. Use these fields to configure conditions that can exclude events. Alternatively, you can filter incoming events to ensure that only those events that meet the required conditions are included.

**Note:** Event rules matching is case-sensitive.

While working in the Event Filter tab:

- In the Event rule list, if you selected a recommended rule or an existing rule, the filter fields are populated with values from these events. You can use these filter fields to compose filter conditions.
- You can click another tab (for example, **Threshold**), work in that area and then return to work in the Event Filter area without losing information.
- Use the right-panel as a source of reference for which fields are available to be used when composing filter conditions.

**Note:** Event filters created on additional_info fields only perform a string comparison and not a numeric comparison, even if the information entered into the value field appears as if it is numeric, as depicted in the following graphic:

1. Navigate to **Event Management > Rules > Event Rules**.
2. Click **New** or open an event rule.
3. Click **Event Filter**.
4. To ignore events, select **Ignore events that match this filter**. All events that match this event rule are ignored. You can add more conditions to the filter by clicking either **OR** or **AND**.

**Note:** Even if an event is ignored by an event rule, the event is still recorded in the Event (em_event) table. Select this option to inhibit an alert from being created and be associated with the event.

5. To configure conditions to which the event rule must apply, click **New Criteria**. You can add further conditions by clicking either **OR** or **AND**.

6. You can add field names to the Event Rule Simple Field Black Lists (em_event_rule_black_list) table to either show or hide fields from the filter.

   - To remove fields from the filter and the right panel, add a new black list field with **Show in rule** not selected.
   - To remove fields only from the filter and still enable it to appear in the right panel, add a black list field with **Show in rule** selected.

7. Click either **Save** or **Submit**.

   Example showing filter conditions that are specified to run a rule when the metric name starts with the word "disk". For this example, any resource is
acceptable, as long as it is not "cpu" and the node name is "MyServer".

Example to filter events using time conditions. In the Select Field or add New field, select Time of event.

Specify the required time conditions, for example, anytime after yesterday.
Find events that are not matched to rules
Find events that are not matched to any rules, and determine if it is necessary to create event rules to manage them.

Role required: evt_mgmt_admin

This video describes event rules and how to find events that are not matched to rules.

2. Near the top of the form, click the link for events or grouped events that are not mapped to rules.
Create an event rule. For example, event rules are useful for managing events that occur regularly.

View event rules
You can view all event rules on the Event Rules list.

Role required: evt_mgmt_admin

You can learn about event rules from the following video tutorial.

Navigate to Event Management > Rules > Event Rules.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The event rule name.</td>
</tr>
<tr>
<td>Override default binding</td>
<td>Legacy binding criteria. If, in the Bind section criteria have not been specified or if the specified bind criteria are not matched, then the legacy binding criteria are considered.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which an event rule is evaluated when multiple rules are defined for the same type of event. Event rules are evaluated in ascending order.</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Updated</td>
<td>Update date and time for the rule.</td>
</tr>
</tbody>
</table>

**Note:** You can filter the Event Rule list to display the required subset of the information. However, if you create a favorite link after filtering, when the link is clicked the Event Rule list does not display the correct filter.

Find rules that were applied to events or alerts
View the rules that were applied to events and alerts to confirm how events are processed.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

Do one of the following:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Find rules that apply to an event | 1. Navigate to Event Management > All Events, and then click an event Number.  
2. Under Related Links, click Check process of event.  
3. Review the information that appears above the Source field, and review all Processing Notes. |
| Find rules that apply to an alert | 1. Navigate to Event Management > All Alerts, and then click an alert number.  
2. In the processing notes, review information about the applied event rule. |

**Configure an event rule to customize alert content**
You can configure an event rule to customize alert content. You can customize the order of the fields and select which fields display. The fields in the left-hand workarea of the Transform and Compose Alert Output section of an event rule are the fields that appear in the generated alert.

Role required: evt_mgmt_admin

You can customize alert content by adding the required event fields and modifying the text in the fields.

**Note:** When creating regex expression for input, ensure that you create input with the variables in sequential order. While variables can be skipped for inclusion, the selected order must be maintained.

Format alert text by using one or both of these methods:

**Format alert text**

<table>
<thead>
<tr>
<th>Format method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transform</td>
<td>Create a field to display in an alert. The field and its value appear in the Additional Information section of an alert. In the Event Input pane, the new field is listed in the Expressions section.</td>
</tr>
<tr>
<td>Compose</td>
<td>Determine which text to display in the selected field.</td>
</tr>
</tbody>
</table>

**Note:**
- Configure which fields display in the Event Input pane by configuring the blacklist (em_event_rule_black_list).
  A field entered into the blacklist that already appears in an existing event rule is not removed from the rule; instead, it is moved to the Manual Attributes section in the rule.
- Control the order of event form fields by configuring em_event_rule_order.

2. Click New or open an event rule.
3. Click **Transform and compose alert output**. In this section, you can use the **Transform** mode and the **Compose** mode.

4. To transform information in fields—standard mode:

   ![Note: In this step, the selection of text is made in standard edit mode. In this mode, undo is available.](image)

1. Click the required field in the Event Input pane. The value and contents of the selected field appear in the workarea on the left.

2. Select standard edit mode.

3. Select the required text. You can double-click text in the field to select a word or select the required text and symbols.
4. Select a field name from the list or enter the required expression name. The field name appears in the Expressions pane and the selected text is highlighted. Both the text and the expression name are highlighted in the same color. Each additional expression that you create, plus its value, is highlighted in a different color.

Note: Transform field names cannot contain a ` (single quote) character.

5. Click Done. The expression name appears in the Expressions section of the Event Input pane.

6. If you click an expression or its value in the Event Input pane, the details display in the workarea on the left.

5. To transform information in fields—regex mode:

Note:
- In regex mode, undo is not available.
- When building a regex expression from variables that are highlighted in the event, place the selected variables in the same order that they appear in the event field.
- When composing the event rule regex, specify the terms using an exact match of the input from the event fields. For example, for the text vhxy2stlgwc_8888_hec03v012979, the regex
  
  ^([^_]+)_.

  looks as if it should catch the text “vhxy2stlgwc” that appears before the underscore. However, it does not work as this text is not an exact match. Use the full regex, as follows:

  ^([^_]+)_.*

- Java flavor is the only regex engine in use.
- The Java regex engine “matches” method is used. With this implementation, the regex engine attempts to match the entire input sequence against the specified pattern. Other matching methods, for example, the find method, are not supported.
- No regex flags are set by default, except the Pattern.DOTALL flag. With this implementation of Java regex, the DOT character `.` matches any character, including a line terminator.
- Aside from Pattern.DOTALL, other regex flags cannot be changed.
1. Click the required field in the Event Input pane. The value and contents of the selected field appears in the workarea on the left.

2. Select regex edit mode.

3. Select the required text and symbols. Replace the highlighted text with (.*) to create a regex expression.

4. In the Expressions pane, click the Select or Add Field field and select a field name from the list or enter the required expression name.

5. Click Done.

Note:
- When building a regex expression from variables that are highlighted in the event, place the selected variables in the same order that they appear in the event field.
- When composing the event rule regex, specify the terms using an exact match of the input from the event fields. For example, for the text
To compose text, in the Transform and compose alert output section, click in the required field.

**Transform and compose alert output**

Click Event Raw values to create new regex expressions.

<table>
<thead>
<tr>
<th>Description</th>
<th>${description}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>${type}</td>
</tr>
</tbody>
</table>

Compose text in the alert fields by performing any of these actions:

- Drag a field from the Event Info pane into the required field.
- Use the Event Info pane as a reference of which fields are available. Write the field name details, using the format $\{<fieldname>\}$, where "$\$" means "use the value". Between the brackets, the name of the event field replaces the $\{fieldname\}$ variable.
- Freely enter text, spaces, or characters. For example, use spaces or underscore symbols (_) to make the text in the alert field readable.

7. Select Manual attributes to specify fields that must display in the generated alert. These fields, and their values, display in the Additional Information section of the generated alert.

8. Click Save or Update.

You can filter event rules. See [Filter the events that an event rule applies to](#).

**Set a threshold to suppress alert generation**

The event threshold is the rate upon which Event Management generates an alert. Receiving multiple events for a device over a short interval may warrant creating an alert, as the condition may be serious. However, receiving events over a longer interval may indicate a less serious situation which would not warrant creating an alert.

Role required: evt_mgmt_admin

You can configure the properties in an event rule to create alerts, suppress alert generation, or close existing alerts according to a specified threshold. The threshold is based on the value of event fields or number of occurrences over a specified period.

**Note:** Field Name can be the name of any numeric field in the Additional information field of the event. Therefore, if CPU is an additional information field for a specific event, then CPU can be used as a Field Name.

Assume that you want to generate an alert when CPU utilization reaches or exceeds 80% three times with no more than 20 seconds between any two consecutive events. Create an event rule with these settings (an explanation for each value is given in parentheses):
**Create Alert Operator**: >= (operator to determine whether utilization of **Field Name** reaches or exceeds the specified value)

- **Field Name**: CPU (events regarding high CPU usage)
- **Threshold Value**: 80 (percent)
- **Occurs**: 3 (three events occur where the CPU usage is at or above ”>=” 80%)
- **Over(seconds)**: 20 (20 seconds or less between events)

To demonstrate how the preceding settings are evaluated, assume that the following events are received:

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Outcome</th>
</tr>
</thead>
</table>
| Reported elapsed time and the CPU usage for each event:  
  - First event elapse time 20, CPU=85  
  - Second event elapse time 40, CPU=80  
  - Third event elapse time 60, CPU=70 | In this scenario, no alert is generated since one event has a CPU utilization that is under 80%. |
| Reported elapsed time and the CPU usage for each event:  
  - First event elapse time 20, CPU=85  
  - Second event elapse time 40, CPU=90  
  - Third event elapse time 70, CPU=95 | In this scenario, an alert is not generated since the elapsed time in one event is over the specified 20 seconds. |
| Reported elapsed time and the CPU usage for each event:  
  - First event elapse time 20, CPU=85  
  - Second event elapse time 40, CPU=95  
  - Third event elapse time 60, CPU=90 | In this scenario, an alert is generated since in all events the elapsed time is within the specified time and the CPU usage is over 80%. |

**Note:**

- When configuring an event rule to create or close alerts according to a threshold, events that arrive at the same second, as determined by the `time_of_event` field, are skipped. This occurs because these events are considered to be duplicates.
- When threshold conditions are configured, alerts close when the **Close Alert Operator** condition is met, and not when an event is received with `Severity = Clear` or `Resolution state = Closing`.

1. Navigate to **Event Management > Rules > Event Rules**.
2. Create or open an event rule.
3. Click **Threshold**.
4. Select **Active**. If **Active** is not selected, the fields on this page do not appear.
5. In the **Create Alert Operator** field, select an operator.
   - If you select **Count**, specify the corresponding **Occurs** and **Over(seconds)** fields.
   - If you select any operator other than **Count**, the **Threshold Metric** and **Value** fields appear. Specify the required values in these fields.
Note: The value of the Threshold Value property can be the name of any field in the Additional information of the event. For example, if CPU is a field in Additional information for a specific event, then CPU can be used as the Threshold Value.

- In the Occurs field, specify the required value.
- In the Over(seconds) field, specify the required period.

6. To automatically close alerts, in the Close Alert Operator field, select an operator. Extra fields appear according to your selection.
   - If Count is specified as the operator for the Create Alert Operator field, then the selection in the Close Alert Operator field is either None or Idle. Specify the required value.
   - If you select Idle, configure the Over(seconds) field.
   - In the Close Alert Operator field, if you select an operator other than Idle, then configure the Value, Occurs, and Over(seconds) fields.

7. Click Save or Submit.

To create an alert when a specific event occurs 5 times in 10 minutes, in Threshold:

1. In the Threshold Metric field, specify the name of any field that exists in the Additional information field in the event. The value of the field is irrelevant.
2. In the Create Alert Operator field, select Count.
3. In the Occurs field, specify 5.
4. In the Over field, 600 (10 * 60 seconds).
5. Click Save or Submit.

To create an alert when a specific event occurs 5 times in 10 minutes with a metric value greater than 55. Assume that "metric_value" is a field in the Additional information of the event. Specify:

1. Select the Active check box.
2. In the Create Alert Operator field, select >=.
3. In the Threshold Metric field, specify metric_value.
4. In the Value field, specify 55.
5. In the Occurs field, specify 5.
6. In the Over field, specify 600 (10 * 60 seconds).
7. Click Save or Submit.

Alert binding to CIs with event rules
When alerts are associated with CIs, the task of remediation is simplified. During alert generation, Event Management uses event rules and other mechanisms to automatically bind alerts to CI information from the CMDB. For tracking purposes and remediation, the alert shows information about the CI that caused the event.

Alert binding process flow
Alerts bind to CIs based on the following process flow:
1. When an event arrives, Event Management checks the node or CI identifiers.

2. If no node exists, the generated alert can bind to the CI using the alert Type, Additional information, or Configuration item identifier fields.

3. If the event has a node value, search for a valid host.

4. If the event has a host and a CI type, try to bind to a device CI.

5. If the event has a host, try to bind to the application CI.
How alerts bind to CIs

1. Node or CI Identifiers Exist in Event?
   - Yes: Non-host CI (Service or Group) Binding
   - No: Node Empty but CI Identifiers Exist in Event?

2. CI Type Exists in Event?
   - No: Search for Any CI
   - Yes: CI Found?

3. Host CI Binding
   - Search for Host
     - Host Found?: Yes
     - No: Node or CI Identifiers Exist in Event?

4. Device CI Binding on Host
   - Event Has Device Type?
     - Yes: Search for Device
     - No: No Binding
     - Device Found?: Yes
     - No: Bind to Device On Host

5. Application CI Binding on Host
   - Event Has Process Info or App Type?
     - Yes: Search for Process
     - No: Process Found
     - Yes: Bind to Application CI

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
The event can contain the binding process flow in its *Processing Notes* field.

**Tracking and remediation**

Alerts can be bound to CIs from the CMDB for tracking purposes and remediation. Event Management uses event rules and various mechanisms to automatically bind CIs to alerts. When information from an event populates a field with a value, the value either originates from the event source or from event rules. This enhances remediation, functionality, and integration with other ITOM products.

**Binding to an application running on a specific host**

If the event is specific to an application type, use the following steps to bind alerts to a specific application:

- Use the procedures in the topic.
- Create an event rule with a filter that captures events on the application type you want.
- In the event rule, select **Binding**.

**Binding**

---

Select to bind alert to CI using CI identifiers.

Default binding: Value of Node field will be used to try and match CI names in Host CIs, such as Computer, OS, Switch Router (any CI type extending CI).

- **Override default binding**

**Binding type**

- **CI Identification**

- Click **Override default binding**.
- In the Binding Type field, select either **CI's Identification** or **CI field matching**.
• For **CI’s Identification**, specify the **Class**.

**Binding**

Select to bind alert to CI using CI identifiers.

**Default binding**: Value of Node field will be used to try and match CI for Host CIs, such as Computer, OS, Switch Router (any CI type except for the current CI type).

- **Override default binding**

**Binding type**: CI Identification

**Class**: Tomcat Connector

**Criterion attributes** - Fill at least one set of criterion attributes

- **Select an Identifier** ▼ is Add Value

**Container level 1**: Tomcat

**Criterion attributes** - Fill at least one set of criterion attributes

- **Select an Identifier** ▼ is Add Value

**Container level 2**: Select a Class

• In the **Criterion attributes field**, specify name and sys_class_name.
In the name Add Value field, specify the required name.

In the Container level 1 area, specify the required values.

If further container level fields appear, specify the required values.

For CI field matching, specify the required CI Type.

- In the binding process, after the host is found, the algorithm matches all additional_info attributes that have the same name as CI fields for that CI type. If the match is successful, the event is bound to the CI.
- If more than one matching application is found on the host, the alert is bound to the host and not to the application.

Where there is no CI Type, for example, if you want to bind alerts to a SQL server application when the CPU on sqlServer.exe is over 90%, instead use these procedures:

- Populate the Node field in the event with the Cl name, FQDN, IP, or MAC address value. The bind is successful even if host has more than one IP address or MAC address.
- If you want to use a unique identifier that is not one of the four mentioned above, populate the event rule CI Identifier (ci2metric_id) field with one or more unique identifiers of the CI. This field should be in JSON format. For example, to use a unique identifier that is not one of the four mentioned above, add a CI Identifier (ci2metric_id) filter field with one or more unique identifiers of the CI. If the host CI is VMWare VM, and it has a field called MOID, use the JSON format and specify: {"moid":"<CI moid>"}
- Create an event rule with an event match field which maps the process name to the mapping variable sa_process_name. In this case, do not use the CI type.

Binding procedures

Use the procedures in these topics to bind alerts.

Bind alerts to a host CI

You can create an event rule to bind alerts to host CIs.

Role required: evt_mgmt_admin

An incoming event from a host CI can bind to an alert based on the event Node field value. If the value resides in a different event field, you can use an event rule transform to copy the data to the Node field on the alert.

When binding alerts to a host, such as a computer, switch, or router CI:

- Populate the Node field in the event with the CI name, FQDN, IP, or MAC address value. The bind is successful even if host has more than one IP address or MAC address.
- If you want to use a unique identifier that is not one of the four mentioned above, populate the event rule CI Identifier (ci2metric_id) field with one or more unique identifiers of the CI. This field should be in JSON format. For example, to use a unique identifier that is not one of the four mentioned above, add a CI Identifier (ci2metric_id) filter field with one or more unique
identifiers of the CI. If the host CI is VMWare VM, and it has a field called MOID, use the JSON format and specify: "moid": "<CI moid>")

2. Click New, and in Event Rule Info, fill in the appropriate fields of the event rule.
3. In the Event Filter condition builder, select Node, select an operator, and then enter one of these values: CI name, FQDN, IP, or MAC address.
4. Click Transform and Compose Alert Output and specify any Additional information mappings that you require.
5. Click Submit.

Bind alerts to a specific host CI
You can create an event rule to bind alerts to the correct device CI. An incoming event from a device CI can bind to an alert based on event rule transform information. However, first identify the host CI that the device is running on.
Role required: evt_mgmt_admin

In the binding process, after the host is found, the algorithm matches all additional_info attributes that have the same name as CI fields for that event type. If the match is successful, the event is bound to the CI. If more than one matching device is found on the host, the alert is bound to the host and not the application. Use the following steps to bind alerts to a specific device:

- When binding alerts to a host, such as a computer, switch, or router CI:
  - Populate the Node field in the event with the CI name, FQDN, IP, or MAC address value. The bind is successful even if host has more than one IP address or MAC address.
  - If you want to use a unique identifier that is not one of the four mentioned in the preceding point, populate the event rule CI Identifier (ci2metric_id) field with one or more unique identifiers of the CI. This field should be in JSON format. For example, to use a unique identifier that is not one of the four mentioned above, add a CI Identifier (ci2metric_id) filter field with one or more unique identifiers of the CI. If the host CI is VMWare VM, and it has a field called MOID, use the JSON format and specify: "moid": "<CI moid>"

- Create an event rule with a filter that captures events on the device type you want. In the event rule, select the Transform check box and select the appropriate CI Type:

<table>
<thead>
<tr>
<th>Device to bind</th>
<th>CI Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>File System</td>
<td>cmdb_ci_file_system</td>
</tr>
<tr>
<td>Port</td>
<td>cmdb_ci_network_adapter</td>
</tr>
<tr>
<td>Storage Device</td>
<td>cmdb_ci_storage_device</td>
</tr>
<tr>
<td>Volume</td>
<td>cmdb_ci_storage_volume</td>
</tr>
</tbody>
</table>

2. Click New and fill in the appropriate fields of the event rule.
3. Click Transform.
4. Click Active.
5. From the CI type list, select the device that is used in the filter condition. For example, if you build a filter condition for a port, select Port.

<table>
<thead>
<tr>
<th>Device to bind</th>
<th>CI type</th>
</tr>
</thead>
<tbody>
<tr>
<td>File System</td>
<td>cmdb_ci_file_system</td>
</tr>
<tr>
<td>Port</td>
<td>cmdb_ci_network_adapter</td>
</tr>
</tbody>
</table>
In the Event Match Fields section, specify any **Additional information** mappings.

Click **Submit**.

To bind to custom device types, configure the alert binding to a process and add the custom device type to the Process to CI Type Mapping (em_binding_process_map) table.

Example showing binding to a specific device using CI field matching to bind to the following string values in the **Additional information** field of an event: "file_system":"ext2", "mount_point":"mp1".

1. Navigate to **Event Management > Rules > Event Rules**.
2. Click **New**, and in **Event Rule Info**, fill in the appropriate fields of the event rule.

   **Event Rule Info**

<table>
<thead>
<tr>
<th>Name</th>
<th>test binding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>source1</td>
</tr>
<tr>
<td>Order</td>
<td>100</td>
</tr>
</tbody>
</table>

   **Description**: Example using CI field matching to bind to the following string values in the **Additional information** field of an event: "file_system":"ext2", "mount_point":"mp1".

3. Click **Binding**.
   a. Click **Override default binding**.
   b. In the **Binding type** field, select **CI field matching**.
   c. In the **CI type** field, select **File System**.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
4. Click **Save**.

In the Binding Device Maps (em_binding_device_map) table, ensure that File System is mapped to cmdb_ci_file_system.
In the File Systems (cmdb_ci_file_system) table, ensure that the required values exist, for example, name1 and name2, as depicted in the following image.

![File Systems Table](image)

The result of the binding performed by the 'test binding' event rule is shown in the Processing Notes field of the event.

![Processing Notes](image)

Bind alerts for non-host CIs

An incoming event from a discovered business service, application service, or alert group can bind to an alert based on an event rule and the corresponding event field mapping. The event field mapping requires a URL or the port number and corresponding IP address for each service or alert group.

Role required: evt_mgmt_admin

If the event is specific to a non-host CI, for example a business service, application service, or alert group, use these steps to bind alerts to a non-host CI:

- Leave the **Node** field empty.
- Populate the **CI Type** with the CI type you want to bind.
- Make sure the **additional_info** field has enough information to uniquely identify the CI. The algorithm matches all **additional_info** attributes that have the same name as CI fields for that **Event Type**. If the match is successful, the event will be bound to the CI.

Optional method:
- Leave the **Node** field empty.
- Populate the **CI Identifier** (ci2metric_id) field with attributes, as described above, that uniquely identify the CI.

1. Navigate to **Event Management > Rules > Event Rules**.
2. Click **New** and fill in the appropriate fields of the event rule.
3. Click **Transform**.
4. From the **CI type** list, select **cmdb_ci_service_auto**.
5. In the Event Match Fields section, insert a new row with these parameter values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>node</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>(.*)</td>
</tr>
<tr>
<td>Mapping</td>
<td>temp node</td>
</tr>
</tbody>
</table>

6. In the Event Compose Fields section, insert a new row with these parameter value:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>node</td>
</tr>
<tr>
<td>Composition</td>
<td>(empty)</td>
</tr>
</tbody>
</table>

7. Click **Submit**.
8. Navigate to **Event Management > Rules > Event Field Mapping**.
9. Create the corresponding event field mapping with these parameter values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Specify the event monitor software that generated the event.</td>
</tr>
<tr>
<td>Mapping type</td>
<td>Select <strong>Single field</strong>.</td>
</tr>
<tr>
<td>From field</td>
<td>Specify temp node.</td>
</tr>
<tr>
<td>To field</td>
<td>Specify name.</td>
</tr>
</tbody>
</table>

10. In the Event Mapping Pairs section, insert new rows for each event mapping pair.
   a) Set the **Key** with the URL or an IP address with corresponding port value.
   b) Set the **Value** with the discovered business service, application service, technical service, or alert group name.

For example, you can add an event mapping pair for each business service.
11. Click **Submit**.

Bind alerts for application CIs

An incoming event from an application CI can bind to an alert based on event rule transform information. Create an event rule to bind alerts to the host CI and also to the application CI.

**Role required:** evt_mgmt_admin

For example, you can bind alerts to a SQL server application when the CPU on sqlServer.exe is over 90%:

- Make sure that the event includes the host name.
- Create an event rule with an Event Match Field entry that maps the process name to the `sa_process_name` mapping variable. In this case, do not use the **CI type**.
- Use the Process to CI Type Mapping module to add custom patterns.

In the binding process, the following steps occur:

- Identify the host.
- If the host is found, find the application on the host. An application is considered to be running on the host if it has a 'Runs on: Runs' relationship from the application to the host CI in the cmdb_rel_ci table.
- If the event rule for the transform **CI type** is defined, find an application CI on that host in the CMDB.
- If all the alert **additional_info** about the CI matches CMDB CI fields, bind the alert to the application CI.

1. Navigate to **Event Management > Rules > Event Rules**.
2. Click **New** and fill in the appropriate fields of the event rule.
3. In the **Event Filter** tab, select **Node**, select an operator, and then enter the CI name, FQDN, IP, or MAC address values for the host binding. For example, you can specify that the node is FQDN.
4. Click **Submit**.

**Trigger alerting bind to a CI**

You can manually bind an alert to a CI by triggering a new alert.

**Role required: evt_mgmt_admin**

If an alert did not automatically bind to a CI, you can manually trigger a new alert and rebind:

- Delete the alert and resend the event.
- Close the alert, temporarily set the `evt_mgmt.active_interval` system property time to 0, and then resend the alert. After the bind completes, reset the property back to the previous value.

**Custom alert fields**

You can populate custom alert fields with data contained in **Additional information** field of the event.

Valuable data contained in **Additional information** fields of events can be useful, for example, for reporting. Alert fields are automatically populated from fields that have the same name in the event. This behavior holds true for **Additional information** event fields and for **Additional information** fields that Event Rule adds. Therefore, to populate a custom alert field with the value in **Additional information** fields, use the custom field name in the **Additional information** field. You can also use Event Rules for this purpose. Values in the **Additional information** field of an Event that are not in JSON key/value format are normalized to JSON format when the event is processed.

The short description field of an alert is automatically written using the type, node, and description of the alert. To prevent the short description field from being over-written, open the **sys_properties** table, locate the `evt_mgmt.override_alert_short_description` property, change the value to `false`, and then click **Update**.

Depending on permissions, you may only be able to create fields with the **user_** prefix. If so, use Event Rules to create an **Additional information** field with the same name. To prevent some fields to be copied to the alert field, use the `evt_mgmt.alert_black_list_fields` property and add the field names that must be excluded. By default, the fields that are not copied are:

- `message_key`
- `category`
- `additional_info`
- `sys_updated_on`
- `sys_updated_by`
- `sys_created_by`
- `sys_created_on`
- `sys_mod_count`
- `sys_id`
**View patterns for event group creation**

Event groups are sets of events that do not have a matching event rule. You can view the patterns in a group of events to learn the impact of creating a rule based on the event source and description patterns.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

Events are grouped according to their source and description.

2. Click the event or grouped event link.
3. In the list of events, click an event.
4. To change the pattern of an event, double-click the pattern, make your changes, and then click **Update**.

---

**Event collection configuration**

Event Management receives external events via an event collector or script. If you are using a script to collect events, no configuration is required. All other methods of collecting events require configuration.

After event collection configuration, Event Management can collect events that are sent directly to the instance, or via the MID Server, SNMP trap, or email.

**Standard event form**

A standard event form must be used by all events, regardless of how they arrive at the instance. For more information, see [Event field format for event collection](#).
**Event collection via MID Server**

The MID Server facilitates communication and movement of data between external applications, data sources, services, and the ServiceNow instance.

<table>
<thead>
<tr>
<th>Connector operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pull</td>
<td>You can configure connector instances to use connector definitions (script instructions) that enable the MID Server to connect to external servers to obtain event information using a pull operation. For more information, see <a href="#">Configure event collection that uses the pull operation</a>.</td>
</tr>
<tr>
<td>Push</td>
<td>You can configure listeners to push event information to the instance or MID Server. For example, configure the MID WebService Event Collector to push event messages to the MID Server. For more information, see <a href="#">Event collection to MID Server using web service API</a>.</td>
</tr>
</tbody>
</table>

**Event field format for event collection**

Event Management requires all events to use a standard form, regardless of how they arrive at the instance.

In the application navigation filter, enter `em_event.list`.

**Event fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>em_event.source</code></td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td><code>em_event.node</code></td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td><code>em_event.type</code></td>
<td>Pre-defined event type, such as high CPU, which is used to identify an event record. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td><code>em_event.ci_type</code></td>
<td>Pre-defined definition for a configuration item (CI) that resides in the CMDB that describes a category for hardware, software application, or web service.</td>
</tr>
<tr>
<td><code>em_event.ci_identifier</code></td>
<td>JSON string that represents a configuration item. For example, <code>{&quot;name&quot;:&quot;SAP ORA01&quot;,&quot;type&quot;:&quot;Oracle&quot;}</code>. The CI identifier that generated the event appears in the Additional information field. This field has a maximum length of 1000 digits. Note: Reference pop-ups and click-throughs are hidden by default for read-only fields. For Configuration item and other read only fields, you can optionally change the read-only setting. For more information, see <a href="#">Configure pop-ups on read-only fields</a>.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>em_event.resource</td>
<td>Node resource that is relevant to the event. For example, Disk C, CPU-1, the name of a process, or service. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>metric_name</td>
<td>Unique name to identify the metric data.</td>
</tr>
<tr>
<td>em_event.event_class</td>
<td>If the <code>em_event.node</code> field is not specified, it is mandatory for alerts to be created automatically. Values for the <code>em_event.event_class</code> field originate from either the source generating the events or by event rule. Name of the machine or software that generated the event. For example, SolarWinds on 10.22.33.44. Corresponding field display name is <strong>Source Instance</strong>.</td>
</tr>
<tr>
<td>em_event.message_key</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the <strong>Source, Node, Type, Resource, and Metric Name</strong> field values. This field has a maximum length of 1024 digits.</td>
</tr>
<tr>
<td>em_event.severity</td>
<td>Event severity options are:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Critical</strong>: Immediate action is required. The resource is either not functional or critical problems are imminent.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Major</strong>: Major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Minor</strong>: Partial, non-critical loss of functionality or performance degradation occurred.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Warning</strong>: Attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Info</strong>: An alert is created. The resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Clear</strong>: No action is required. An alert is not created from this event. Existing alerts are closed.</td>
</tr>
<tr>
<td>em_event.resolution_state</td>
<td>If the field is empty, the resolution on corresponding alerts is still pending. Event state from the event source is either <strong>New</strong> or <strong>Closing</strong>.</td>
</tr>
<tr>
<td></td>
<td>- <strong>New</strong>, the resolution on corresponding alerts is open.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Closing</strong> event state closes corresponding alerts.</td>
</tr>
<tr>
<td>em_event.time_of_event</td>
<td>Time that the event occurred in the source system. This field is a GlideDateTime field in UTC or GMT format. This field has a maximum length of 40 digits.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>em_event.state</td>
<td>Current processing state of the event:</td>
</tr>
<tr>
<td>(State)</td>
<td>- <strong>Ready</strong>: Event has been received and is waiting to be processed.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Processed</strong>: Event was successfully processed.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Ignored</strong>: Value is not in use.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error</strong>: Failure occurred while processing the event.</td>
</tr>
<tr>
<td></td>
<td>For example, the event collection method or event Severity is blank.</td>
</tr>
<tr>
<td>em_event.alert</td>
<td>If an alert was created as a result of the event, this field contains the</td>
</tr>
<tr>
<td>(Alert)</td>
<td>unique ID that Event Management generates to identify the alert.</td>
</tr>
<tr>
<td>em_event.description</td>
<td>Reason for event generation. Shows extra details about an issue. For</td>
</tr>
<tr>
<td>(Description)</td>
<td>example, a server stack trace or details from a monitoring tool. This field</td>
</tr>
<tr>
<td></td>
<td>has a maximum length of 4000 digits.</td>
</tr>
<tr>
<td>em_event.additional_info</td>
<td>A JSON string that gives more information about the event. The JSON data is</td>
</tr>
<tr>
<td>(Additional information)</td>
<td>supported for String values only, other value types are not supported. You</td>
</tr>
<tr>
<td></td>
<td>must convert numbers to String values by enclosing them in double quotes.</td>
</tr>
<tr>
<td></td>
<td>For example, this value is not supported: <code>{&quot;CPU&quot;:100}</code> while this value is</td>
</tr>
<tr>
<td></td>
<td>supported: <code>{&quot;CPU&quot;:&quot;100&quot;}</code>. Another example of a valid JSON string is:</td>
</tr>
<tr>
<td></td>
<td>This information can be used for third-party integration or other post-</td>
</tr>
<tr>
<td></td>
<td>alert processing. Values in the <strong>Additional information</strong> field of an Event</td>
</tr>
<tr>
<td></td>
<td>that are not in JSON key/value format are normalized to JSON key/value format</td>
</tr>
<tr>
<td></td>
<td>when the event is processed. For example, assume that the following plain</td>
</tr>
<tr>
<td></td>
<td>text is in the <strong>Additional information</strong> field &quot;Connection instance is</td>
</tr>
<tr>
<td></td>
<td>successful&quot;. When the event is processed, all this plain text becomes one</td>
</tr>
<tr>
<td></td>
<td>JSON string and might not be useful within an alert. In the resultant alert,</td>
</tr>
<tr>
<td></td>
<td>this string is in the <strong>Additional information</strong> field in JSON key/value</td>
</tr>
<tr>
<td></td>
<td>format, containing the data: <code>{&quot;additional_content&quot;: &quot;Connection instance is successful&quot;}</code>.</td>
</tr>
<tr>
<td>processing_notes</td>
<td>Display of the events processing log.</td>
</tr>
</tbody>
</table>

**Connectors and listeners**
Event Management provides many connectors and listeners with your subscription to pull or push events from external devices. Connectors and listeners are also available from the ServiceNow store as well as from third parties. You can also create custom connectors and listeners.
## Provided connectors (pull)

This table lists the connectors that are provided with Event Management. For more information, see [Configure event collection that uses pull operation](#).

**Note:** You can also create [custom pull connectors](#).

**Note:** The AzureConnector connector displays in the base system only when the Operational Intelligence plugin (com.snc.sa.metric) is activated.

<table>
<thead>
<tr>
<th>Connectors</th>
<th>Supported version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPOM</td>
<td>08.60.005</td>
<td>Receive events from the HP Operations Manager (HPOM).</td>
</tr>
<tr>
<td>Hyperic</td>
<td>5.8.4.0</td>
<td>Receive events from the VMware vRealize Hyperic server.</td>
</tr>
<tr>
<td>IBM Netcool</td>
<td>7.3.1</td>
<td>Receive events from IBM Netcool/OMNibus Object Servers and Impact Servers.</td>
</tr>
<tr>
<td>Configure event collection from an Icinga2 connector</td>
<td>2.4.1</td>
<td>Receive events while monitoring your network resources.</td>
</tr>
<tr>
<td>NagiosXI</td>
<td>5.5.2</td>
<td>Receive events from the Nagios XI server.</td>
</tr>
<tr>
<td>NNMi</td>
<td>10.30.653</td>
<td>Receive events while monitoring your network resources.</td>
</tr>
<tr>
<td>OMI</td>
<td>10.01, 10.10, 10.11, 10.12, 10.60, and 10.61</td>
<td>The OMI connector is not supported on versions below version 10.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receive alerts from the HP OMI server.</td>
</tr>
<tr>
<td>Op5</td>
<td>7.3.15</td>
<td>Receive alerts from an OP5 Monitor.</td>
</tr>
<tr>
<td>Opsview</td>
<td>5.4.0</td>
<td>Receive alerts from an Opsview Monitor</td>
</tr>
<tr>
<td>PRTG</td>
<td>18.2.4</td>
<td>Receive alerts from a Paessler PRTG Network Monitor source.</td>
</tr>
<tr>
<td>SAP Solution Manager</td>
<td>7.2 740 SP16</td>
<td>Receive alerts from an SAP Solution Monitor Monitor source.</td>
</tr>
</tbody>
</table>
| SCOM                                     | • 2007 – version 6.1.7221.0  
• 2012 - version 7.1.10226.0  
• 2016 - version 7.2.117190 | Alert collection from the Microsoft System Center Operations Manager (SCOM). |
| SolarWinds                               | • NPM 12.0   
• SAM 6.2.4 | Receive events from the SolarWinds monitor.                                |
<p>| vCenter                                  | 6.5               | Receive events from your VMware vSphere environment.                       |</p>
<table>
<thead>
<tr>
<th>Connectors</th>
<th>Supported version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vRealize</td>
<td>VMware vRealize Operations Manager version 6.5.0</td>
<td>Receive events from the vRealize Operations Log and Event Management servers.</td>
</tr>
<tr>
<td>Zabbix</td>
<td>3.0.3</td>
<td>Receive alerts from the Zabbix server.</td>
</tr>
</tbody>
</table>

**Provided listeners (push)**

This table lists the listeners that are provided with Event Management. Generic listener transform scripts are also provided, one for event collection through the MID Server and the other through the instance.

*Note:* You can also create a custom push listener.

<table>
<thead>
<tr>
<th>Listeners</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td>Use event transform script to process Amazon Web Services (AWS) alarms as events. The event collection is through the instance.</td>
</tr>
<tr>
<td>Azure</td>
<td>Use event transform script to process Azure platform event messages. The event collection is through the instance.</td>
</tr>
<tr>
<td>BMC TrueSight</td>
<td>Collect JSON formatted event messages sent from BMC TrueSight Operations Management (TrueSight). The event collection is through the MID Server.</td>
</tr>
<tr>
<td>SNMP traps</td>
<td>Process SNMP traps as events.</td>
</tr>
<tr>
<td>Email</td>
<td>Configure an inbound email action to send email notifications when events and alerts are triggered.</td>
</tr>
<tr>
<td>Oracle Enterprise Manager trap collection</td>
<td>Configure the SNMP trap listener to receive Oracle Enterprise Manager Cloud Control SNMP traps. Event Management can process these traps as events.</td>
</tr>
<tr>
<td>URL format for generic JSON</td>
<td>This URL can be used as-is and requires an event rule to be configured.</td>
</tr>
<tr>
<td></td>
<td>Configure this generic JSON target URL and configure an event rule to collect events: https://&lt;&lt;INSTANCE&gt;&gt;/api/global/em/inbound_event?source=genericJson</td>
</tr>
<tr>
<td></td>
<td>For more information about how to collect events using a MID Server and for an example that shows how to send a generic format and use the event rule designer to create an alert, including the mapping.</td>
</tr>
</tbody>
</table>

**ServiceNow store - connectors and listeners**

Connectors and listeners are available from the ServiceNow store. Download the required content from ServiceNow Store.
Third party - connectors and listeners

Connectors and listeners are available from third parties. For further information, contact your sales representative for available connectors and listeners.

<table>
<thead>
<tr>
<th>Third party connectors and listeners</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>AppDynamics Service Model Integration</td>
<td><a href="#">AppDynamics Service Model Integration</a></td>
</tr>
<tr>
<td>Use this integration to enable your instance to have AppDynamics Application Performance Monitoring data fed into Event Management.</td>
<td></td>
</tr>
<tr>
<td>CA APM Connector</td>
<td><a href="#">CA APM Connector</a></td>
</tr>
<tr>
<td>Use this integration to transfer alerts from CA APM to Event Management.</td>
<td></td>
</tr>
<tr>
<td>CA UIM Connector</td>
<td><a href="#">CA UIM Connector</a></td>
</tr>
<tr>
<td>Use this integration to synchronize alarms from CA UIM into Event Management.</td>
<td></td>
</tr>
<tr>
<td>Datadog Integration</td>
<td><a href="#">Integrate Datadog as a data source</a></td>
</tr>
<tr>
<td>You can create ServiceNow tickets from alarms that are triggered in Datadog. Graphs and comments generated by Datadog can be added to ServiceNow tickets, and you can manage the resolution workflow from within Datadog.</td>
<td></td>
</tr>
<tr>
<td>Dynatrace Integration: AppMon Connector</td>
<td><a href="#">Dynatrace AppMon Connector</a></td>
</tr>
<tr>
<td>Integrate with this application for one-directional transfer of events from Dynatrace to Event Management.</td>
<td></td>
</tr>
<tr>
<td>Dynatrace Integration: Incidents and CMDB Integration</td>
<td><a href="#">Dynatrace Incidents and CMDB Integration</a></td>
</tr>
<tr>
<td>Integrate with this application for one-directional transfer of events from Dynatrace to Event Management.</td>
<td></td>
</tr>
<tr>
<td>HPE SiteScope Integration</td>
<td><a href="#">SiteScope Connector</a></td>
</tr>
<tr>
<td>Integrate with the SiteScope Connector application for the transfer of alerts to Event Management in an automated manner.</td>
<td></td>
</tr>
<tr>
<td>NewRelic Integration</td>
<td><a href="#">NewRelic Integration</a></td>
</tr>
<tr>
<td>Integrate with this application for one-directional transfer of events from NewRelic to Event Management.</td>
<td></td>
</tr>
<tr>
<td>Panopta Integration</td>
<td><a href="#">Panopta integration</a></td>
</tr>
<tr>
<td>Integrate with this application for one-directional transfer of events from Panopta to Event Management.</td>
<td></td>
</tr>
</tbody>
</table>
### Third party connectors and listeners
<table>
<thead>
<tr>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Splunk Integration</td>
</tr>
</tbody>
</table>

Integrate the Splunk Add-on with ServiceNow to enable Splunk users to create incidents and events in Event Management. For more information about the configuration of this integration, see [Splunk Add-on for ServiceNow](#).

### Configure a connector instance
Configure a connector instance to schedule the frequency of event collection.

Before starting this procedure:

- Locate or define a connector definition.
- Create the credentials to connect to the event source.
- If the connector instance is to be used for the collection of Operational Intelligence metric data, the Operational Intelligence plugin must be activated.

**Role required: evt_mgmt_admin**

You can use a connector instance to control the location and manner in which events arrive from external sources. You can optionally select to collect Operational Intelligence information from the external sources that you are connecting to. The Operational Intelligence feature is for the analysis of trends, threshold definitions, and anomaly detection. You can run a test to confirm that connector instance parameters let the MID Server receive events from a supported external event source to the Event (em_event) table. The test also confirms:

- The MID Server is running.
- The host is running on the IP address that is in the **Host IP** field.
- Both the MID Server and the host are running in the same domain.
- The connector instance value fields are valid.
- A connection can be made to the event source, using API, to retrieve events.

1. Navigate to **Event Management** > **Event Connectors (Pull)** > **Connector Instances**.
2. Click **New**.
3. Fill in the fields, as appropriate.

### Connector Instance Form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the connector instance record, such as the name of event source host.</td>
</tr>
<tr>
<td>Description</td>
<td>Any optional information that the administrator wants to use to identify this record.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the connector definition that matches the source of external events. The options available in the base system are, for example:</td>
</tr>
<tr>
<td></td>
<td>- AWS CloudWatch</td>
</tr>
<tr>
<td></td>
<td>- AzureConnector</td>
</tr>
<tr>
<td></td>
<td>- HPOM</td>
</tr>
<tr>
<td></td>
<td>- IBM Netcool</td>
</tr>
<tr>
<td></td>
<td>- NagiosXI</td>
</tr>
<tr>
<td></td>
<td>- NNMi</td>
</tr>
<tr>
<td></td>
<td>- OMi</td>
</tr>
<tr>
<td></td>
<td>- SCOM</td>
</tr>
<tr>
<td></td>
<td>- SolarWinds</td>
</tr>
<tr>
<td></td>
<td>- Op5</td>
</tr>
<tr>
<td></td>
<td>- PRTG</td>
</tr>
<tr>
<td></td>
<td>- vCenter</td>
</tr>
<tr>
<td></td>
<td>- Zabbix</td>
</tr>
<tr>
<td></td>
<td>For a list of the connector definitions available in the base system, see <a href="#">Connectors and listeners</a>.</td>
</tr>
<tr>
<td>Host IP</td>
<td>The IP address of the event source host. The system uses this IP address to select the appropriate MID Server for communicating with the event source host.</td>
</tr>
<tr>
<td>Credential</td>
<td>The record from the Credentials (discovery_credentials) table containing valid credentials to the event source host.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The date and time of the most recent event import. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The status of the last import. The value of this field is automatically populated.</td>
</tr>
<tr>
<td></td>
<td>- None - A valid connection has not yet been established.</td>
</tr>
<tr>
<td></td>
<td>- Success - A successful connection was established.</td>
</tr>
<tr>
<td></td>
<td>- Error - A connection was established. However the external event source was not updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from the external event source. This value cannot be lower than the minimum schedule property, which by default is 120 seconds.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message received by the connector. If the test connector fails, an error message appears in this field.</td>
</tr>
<tr>
<td>Active</td>
<td>This option appears only after the form has been saved. Do not select this check box until after you test the connection between the MID Server and the connector. Select this check box to enable pulling events from this external event source.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Bi-directional</td>
<td>Select to invoke the bi-directional option. This option is available only when the specified connector definition has been configured with bi-directional values. When selected, this option enables the bi-directional exchange of values to-and-from the external event source. The <strong>Last bi-directional status</strong> option appears only when this option is selected. This option appears only if the connector definition that is selected is SCOM.</td>
</tr>
</tbody>
</table>
| Last bi-directional status             | The value of this field is automatically populated. The status values are:  
  - **None** - A valid connection has not yet been established.  
  - **Success** - A successful connection was established.  
  - **Error** - A connection was established. However the external event source was not updated.  
  This option appears only if the connector definition that is selected is SCOM. |
| Metrics collection                     | Designates collection of Operational Intelligence metric data.  
  **Note:** The metrics connector supports working against the SCOM database (MSSQL) that is configured to work with SSL.  
  This option appears only if the connector definition supports metric collection, such as SCOM. |
| Metrics collection last run time       | The date and time of the most recent metric data collection. The value of this field is automatically populated.                                                                                                                                                        |
| Last metrics collection status         | Status of the metric data collection activity. The value of this field is automatically populated.  
  This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and if the **Metrics collection** option is selected.                                                             |
| Metrics collection schedule (seconds)  | The time, in seconds, to repeat the metric data collection scheduled job.  
  This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and when the **Metrics collection** option is selected.                                                                 |
| Metrics database host                  | Enter either the IP address or the host name of the metrics database host.  
  This option appears only if the connector definition that is selected supports metric collection (such as SCOM).                                                                                             |
| Connect using a named instance         | Select to specify the MSSQL named instance to connect to.  
  This option appears only if the connector definition that is selected is SCOM.                                                                                                                              |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics database named instance</td>
<td>The name of the metrics database instance. This option appears only if the connector definition that is selected supports metric collection (such as SCOM).</td>
</tr>
<tr>
<td>Metrics database port</td>
<td>The port used by the metric database. The connection is made using JDBC. The base system value is 1,433. This option appears only if the connector definition that is selected supports metric collection (such as SCOM).</td>
</tr>
<tr>
<td>Database login with Windows authentication</td>
<td>Perform database login with the credentials of the logon user that is defined on the MID Server service. This option appears only if the connector definition that is selected is SCOM.</td>
</tr>
<tr>
<td>Metrics collection running</td>
<td>When selected, indicates that the Operational Intelligence collection is running. This option appears only if the connector definition that is selected supports metric collection (such as SCOM).</td>
</tr>
<tr>
<td>Metrics database credential</td>
<td>Credentials for the metric database. This option appears only if the connector definition that is selected supports metric collection (such as SCOM).</td>
</tr>
<tr>
<td>Connector Instance Values</td>
<td>The related list containing connection parameters for the event source host. The list of parameters depends on the selected connector definition.</td>
</tr>
<tr>
<td>MID Servers for Connectors</td>
<td>The name of the MID Server to process events from the event server.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select **Save**.
5. Click **Test connector** to verify that the MID Server can communicate with the external server host.
6. If the test fails, follow the instructions on the page to correct the problem and then run another test. The description of the reason for failure is shown in the MID Server log, according to the type of error, such as Events, Metrics, or Bi-Directional.
7. After a successful test, select the **Active** check box and then click **Update**.
8. Confirm that Event Management processes events using this connector instance by manually creating an event.

Create a connector definition
Each connector definition is specific to an event source vendor. The connector definition specifies the MID Server script include that pulls events from the external event source. In addition, the connector definition specifies what connector instance value parameters are needed to connect to the external event source host.

Write custom JavaScript code that accomplishes these actions:
- Connect to an event monitoring tool.
- Retrieve events from an event monitoring tool.
- Send events to the Event (em_event) table using the web service API.

Role required: evt_mgmt_admin or evt_mgmt_operator
Create a connector definition to support a new external event source from which to receive events. This procedure automatically creates a JavaScript code file using the name that you enter in the **Name** field as the filename.

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Definitions**.
2. Click **New**.
3. In the **Name** field, type a descriptive name for the connector definition.
4. In the **Script type** field, select **Javascript**.
5. Right-click the form header and select **Save**.
6. The **Javascript to run** field is automatically populated with the name of the JavaScript template file. You can optionally replace the default JavaScript with your JavaScript code, see *Create a custom pull connector*. To edit the default JavaScript file, click the exclamation symbol to the right of the JavaScript file name and follow these steps:
   a) Specify the information required for the **testConnection** function. This function tests the connector definition to verify if the connection to the target is valid.
   b) Specify the information required for the **execute** function. This function retrieves the information from the external source.
   c) Specify the information required for the **retrieveKpi** function. This function retrieves the metric data from the external source.
7. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Connector Definition form</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Field</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>Default schedule</td>
<td>The number of seconds between attempts to receive events.</td>
</tr>
<tr>
<td>Bi-directional</td>
<td>Specify to enable the external monitoring system to be updated if the alert is changed. There is default implementation for SCOM.</td>
</tr>
<tr>
<td>Alert field identifier</td>
<td>Specify the alert field that, if changed manually, causes the external monitoring system to be updated.</td>
</tr>
<tr>
<td>Collect metrics</td>
<td>Select to enable the collection of metrics.</td>
</tr>
<tr>
<td>Metrics collection default schedule (seconds)</td>
<td>Specify the number of seconds of information that must be collected. The default is 10 seconds.</td>
</tr>
<tr>
<td>Connector Parameters</td>
<td>Specify the parameters to enable communication with an event server.</td>
</tr>
</tbody>
</table>
Field | Description
--- | ---
Connector Definition to MID Server Capabilities | The MID Server name to process events from the event server. If not specified, an available MID Server is used.

8. Click **Submit**.

Configure a default MID Server for connectors
You can set a default MID Server for connectors to ensure that there is always a MID Server available to receive external events.

Role required: evt_mgmt_operator

The default MID Server must have connectivity to all external event sources.

1. Enter `sys_properties.list` in the navigation filter to open the System Property (`sys_properties`) table.
2. Select the `mid.server.connector_default` system property.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| mid.server.connector_default | Default MID Server for connectors  
Determine the MID Server connectors to use when no MID Server is specified. Must match a MID Server name.  
- **Type**: select **string** from the list  
- **Value**: enter the name of an existing MID Server, for example, SNC MID Server  
- **Location**: System Property (`sys_properties`) table |

**Note:** You can reset the default MID Server by clearing the `mid.server.connector_default` property or by updating the value of this property to a different MID Server. This property is not automatically reset in all scenarios.

4. Click **Update**.

Legacy: Configure a MID Server for RCA for business services
Configuring MID Servers for Event Management root cause analysis (RCA) for business services is not supported for new instances in the New York release. However, instances upgraded from a previous release that use MID Servers for RCA can continue to do so.

Role required: admin

If the Domain Support - Domain Extensions Installer plugin is activated, then you can configure a dedicated MID Server with the RCA for business services capability per domain. In this case, RCA for a business service is done on the MID Server that is in the same domain as the business service. Otherwise, a MID Server from the global domain is used.
Hardware requirements

<table>
<thead>
<tr>
<th>Component</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td></td>
</tr>
<tr>
<td>Requirement</td>
<td></td>
</tr>
<tr>
<td>Memory</td>
<td>• Minimum: 4 GB</td>
</tr>
<tr>
<td></td>
<td>• Recommended: 8 GB</td>
</tr>
<tr>
<td>Processor</td>
<td></td>
</tr>
<tr>
<td>Requirement</td>
<td>• Minimum: Either of the following:</td>
</tr>
<tr>
<td></td>
<td>• Core 2+</td>
</tr>
<tr>
<td></td>
<td>• Xeon processor with a speed over 2 GHz</td>
</tr>
<tr>
<td></td>
<td>• Recommended: Quad-core</td>
</tr>
<tr>
<td>Disk space (for 100 discovered services)</td>
<td>• Minimum: 8 GB</td>
</tr>
<tr>
<td></td>
<td>• Recommended: 10 GB</td>
</tr>
</tbody>
</table>

Software requirements

<table>
<thead>
<tr>
<th>OS</th>
<th>Supported OS versions</th>
<th>Additional requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>32-bit and 64-bit versions:</td>
<td>For either version of the OS:</td>
</tr>
<tr>
<td></td>
<td>• Windows 2008 R2</td>
<td>• 32-bit version of the MID Server</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2012 R2</td>
<td>• 32-bit version of Visual C++ Redistributable Packages for Visual Studio 2013, which you can download from the Microsoft website download center <a href="http://microsoft.com">microsoft.com</a>.</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2016</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td>• Red Hat Enterprise Edition Linux 6.6 or later</td>
<td>32-bit or 64-bit version of the MID Server</td>
</tr>
<tr>
<td></td>
<td>• CentOS Linux 6.6 or later</td>
<td></td>
</tr>
</tbody>
</table>

Role required: admin

To enable the Event Management root cause analysis (RCA) for business services capability, configure at least one dedicated MID Server with the ServiceAnalytics as a supported application, and with the RCA capability. To ensure uninterrupted services, you can configure a fail-over MID Server cluster for RCA.

The RCA for business services MID Server supports the Event Management RCA for business services process by hosting the RCA Learner and the real-time RCA query handler. On the RCA for business services MID Server, the system builds models that support responses to root cause queries. The learner data on the RCA for business services MID Server persists by default for a maximum of 90 days.

By default, the MID Server is set up as a dedicated MID Server for Event Management RCA for business services, and an attempt to add additional supported applications to the same MID Server is prevented. You should implement Operational Intelligence with its own dedicated Operational Intelligence MID Server. This ensures a high level of performance. However, if needed, you can modify this default behavior and configure the RCA for business services MID Server or the Operational Intelligence MID Server with additional supported applications. For information
about modifying the behavior of the ALL option when selecting supported applications, see Select applications to include in the definition of All for a MID Server.

1. Ensure that the MID Server is validated. For more information, see Validate the MID Server.
2. Navigate to MID Server > Servers.
3. Double-click the MID Server that you want to configure as an RCA for business services MID Server.
4. Add the ServiceAnalytics application:
   a) At the center of the MID Server form, click Supported Applications.
   b) In the Supported Applications section click Edit.
   c) In the slushbucket select ServiceAnalytics and click the > add button.
   d) click Save.

If the MID Server needs to support ServiceAnalytics as well as one or more other applications, you can modify the definition of the ALL option to include these applications, and then select ALL in the slushbucket. ALL is the only option to which it is valid to add the ServiceAnalytics option. Performance might be compromised with these settings.

5. Add the RCA capability:
   a) At the center of the MID Server form, click Capabilities.
   b) In the Capabilities section click Edit.
   c) In the slushbucket select RCA and click the ‘>’ add button.
   d) click Save.

6. Click Update.

The RCA for business services MID Server is automatically configured with a Discovery IP range set to 0.0.0.0-0.0.0.0.

Create a failover cluster for RCA. In this cluster, add the MID Servers that were configured for RCA for business services. For more information, see Configure a MID Server cluster.

Configure event collection that uses pull operation
Event Management provides default connector instances to use connector definitions (script instructions).

Configure the default connectors to connect to external servers to obtain event information using a pull operation.

Event Management and Operational Intelligence share connector definitions and connector instances to pull events and metric data from the same data source. Therefore, if a connector instance for a specific data source already exists, you can configure that connector instance to also obtain event information.

Configure event collection from HP OMI
Configure the HP Operations Manager i (OMI) connector instance to receive alerts from the HP OMI server.

Supported versions: 10.01, 10.10, 10.11, 10.12, 10.60, and 10.61

The OMI connector is not supported on versions below version 10.

Create Basic Auth type of credential to connect to OMI.

Role required: evt_mgmt_admin

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click **New** and create a connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the OMi connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the OMi event collection instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the OMi IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Click in the <strong>Credential</strong> field. Select a Basic Auth credential from the list or click the search icon. Either select the required credentials from the list or click <strong>New</strong> and create the required credentials using Basic Auth type. If you create the credentials, save them using a unique and recognizable name, for example OMIOPS.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>Last run time value. Automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>Last run time status. Automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>Frequency in seconds that the system checks for new events from OMi. The default value is 120 seconds.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message. Automatically updated.</td>
</tr>
<tr>
<td>Connector Definition</td>
<td>Select <strong>OMi</strong>.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select **Save**.
The connector instance values are added to the form and the parameters that are relevant to the connector appear.
4. In the Connector Instance Values section, specify the OMi values.
   a) **days_from.** Specify the number of days for which events must be collected on the first collection cycle. Default 7.
   b) **debug.** Default false. To enter debug mode, specify true.
   c) **port.** Default 80.
   d) **protocol.** Default http.
   e) **query_filter.** Default value is false. If you do want to use this extra filter to query events, specify the event attributes. For example, `assigned_user = "admin" AND title = "My Title"`.

The query filter criteria are:
- A filter property that specifies an event attribute, such as, `related_ci`
- A supported operator, for example, `OR` or `AND`
- A value for the filter property, for example, `admin` or `integrator`

Everything from the `query_filter` field undergoes URL encoding and is then passed as a parameter to the OMi API call. For detailed information on how to build the filter query, see “Filtering by Event Attributes: query” in the HP OMi Extensibility Guide. For example, to query the events for a list of events that are assigned to the admin user, specify:

```
event_list?query=assigned_user%20EQ%20"admin"
```

To nest `query_filter` event attributes, specify the hierarchy using square brackets `( )`. 
f) **whitelist.** Default value: `false`. Specify a comma-separated list to include this feature to add attributes that should be collected and added to the Event Management **Additional information** field.

**5.** Right-click the form header and select **Save**.
**6.** Click **Test Connector** to verify the connection between the MID Server and the connector.
**7.** If the test fails, follow the instructions that the error issues to correct the problem and then run another test.

Note: Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the HP OMi server.

**8.** After a successful test, select the **Active** check box and then click **Update**.

Configure event collection from HPOM

Configure the HPOM connector instance to receive events from HP Operations Manager (HPOM).

Role required: `evt_mgmt_admin`

Supported version: **08.60.005**

**1.** Navigate to **Event Management > Event Connectors (Pull) > Connector Instances**.
**2.** Click **New** and create an HPOM connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a unique name for the HPOM connector instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the HPOM IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Click in the <strong>Credential</strong> field. Select a credential from the list or click the search icon. Either select the required credentials from the list or click <strong>New</strong> and create the required credentials. If you create the credentials, save them using a unique and recognizable name, for example, HPOMOPS.</td>
</tr>
<tr>
<td>Schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from HPOM.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for the use of the HPOM event collection instance.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the <strong>HPOM</strong> connector definition.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last run status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

**3.** Right-click the form header and click **Save**.

The connector instance values are added to the form and the parameters that are relevant to the connector appear.
4. In the Connector Parameters section, specify the values of the mandatory HPOM parameters.
   - **driver.** Specify the driver to be used to make the call. This driver is specific to the type of database you are calling into.
     - For an IBM DB2 Universal driver, specify: `com.ibm.db2.jcc.DB2Driver`
     - For an Oracle driver, specify: `oracle.jdbc.OracleDriver`
     - For a Microsoft SQL driver, specify: `com.microsoft.sqlserver.jdbc.SQLServerDriver`
     - For a MySQL driver, specify: `com.mysql.jdbc.Driver`
     - For a Sybase driver, specify: `com.sybase.jdbc3.jdbc.SybDriver`
   - **url.** Specify a URL. The JDBC protocol uses a connection string to establish the authentication and other parameters to establish a connection between the client and the database. Each database has its own connection string format. In the following URLs, replace the variables, for example, `<username>` and `<password>`, with your values.
     - For an IBM DB2 Universal URL, specify: `jdbc:db2://<IP address>:<port number>/<database name>:user=<username>;password=<password>`
       For example: `jdbc:db2://sysmvs1.stl.ibm.com:5021/regionsdb:user=example;password=w1rs2`
     - For an Oracle URL, specify: `jdbc:oracle:thin:@<IP address>:<port number>:<database name>`
       For example: `jdbc:oracle:thin:@172.31.255.255:4028:OracleMain`
     - For a Microsoft SQL URL, specify: `jdbc:sqlserver://<IP address>;user=<username>;password=<password>`
       For example: `jdbc:sqlserver://localhost;user=example;password=w3lrs2`
     - For a MySQL URL, specify: `jdbc:mysql://<IP address>/database?user=<username>;password=<password>`
       For example: `jdbc:mysql://localhost/database?user=example;password=65wlrs`
     - For a Sybase URL, specify: `jdbc:sybase:Tds:<IP address>:<port number>/<database name>?user=<username>;password=<password>`
       For example: `jdbc:sybase:Tds:172.31.255.255:4100/SYBMAIN?user=example;password=w1rs1m`

5. Use a network tool, such as ping, to verify that the HPOM server is running and the IP address matches the value in the **Host IP** field.

6. Click **Test connector** to verify the connection between the MID Server and the HPOM connector.

7. If the test fails, follow the instructions that the error issues to correct the problem and then run another test.

   **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the HPOM server.

8. After a successful test, select **Active** and then click **Update**.

Configure event collection from Hyperic
Configure the Hyperic connector instance to receive events from the VMware vRealize Hyperic server.

Role required: **evt_mgmt_admin**

Supported version: 5.8.4.0

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances.**
2. Click **New** and create a Hyperic connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a unique name for the Hyperic connector instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the Hyperic IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>In the Credentials form, click <strong>New</strong> and create the required credentials. Save the credentials using a unique and recognizable name, for example, HypericOPS.</td>
</tr>
<tr>
<td>Schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from Hyperic.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for the use of the Hyperic event collection instance.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the <strong>Hyperic</strong> connector definition.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last run status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select **Save**. The connector instance values are added to the form and the parameters that are relevant to the connector appear.

4. In the Connector Parameters section, specify the values of the mandatory Hyperic parameters.
   a) **port** Default value 7080.
   b) **protocol** Default value http.

5. Click **Test connector** to verify the connection between the MID Server and the Hyperic connector.

6. If the test fails, follow the instructions that are issued by the error to correct the problem and then run another test.

   **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the Hyperic server.

7. After a successful test, select the **Active** check box and then click **Update**.

Configure event collection from IBM Netcool

Configure the IBM Netcool connector to receive events from IBM Netcool/OMNIbus Object Servers and Impact Servers.


© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Role required: evt_mgmt_admin

Connector validation performed on IBM NetCool/OMNIbus version 7.3.1

To use the Netcool connector, configure a connector instance for the MID Server and add the jconn3-1.0.jar file that contains the necessary Netcool Sybase jar file.

1. Navigate to MID Server > JAR Files.
2. Click New.
3. Fill in the Name and Description fields.
4. Attach the file that was uploaded in the Before you begin section:
   a) Click the attachments icon ( ).
   b) Click Choose Files and navigate to the temporary download directory.
   c) Select the jconn3.jar file and click Open.
   d) Close the pop-up window to return to the form.

5. Click Submit.
7. Click New and create a connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a unique name for the Netcool connector instance.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for the use of the Netcool event collection instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the Netcool IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>In the Credentials form, click New and create the required credentials. Save the credentials using a unique and recognizable name, for example, NetcoolOPS.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from Netcool. The default value is 120 seconds.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the IBM Netcool connector definition.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a unique name for the Netcool connector instance.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for the use of the Netcool event collection instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the Netcool IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>In the Credentials form, click <strong>New</strong> and create the required credentials. Save the credentials using a unique and recognizable name, for example, NetcoolOPS.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from Netcool. The default value is 120 seconds.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the <strong>IBM Netcool</strong> connector definition.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Specify a MID Server that is up and is valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section. If no MID Server is specified, an available MID Server that has a matching IP range is used.</td>
</tr>
</tbody>
</table>

8. Right-click the form header and click **Save**. The connector instance values are added to the form and the parameters that are relevant to the connector appear.

9. In the Connector Parameters section, specify the value of the mandatory Netcool parameter, **url**. Build a URL using the provided format.

   For example: `jdbc:sybase:Tds:10.11.15.118:4100/NCOMS`

**Note:** After updating the **Host IP** field, you must update the URL parameter, as it does not update automatically.
10. Use a network tool, such as ping, to verify that the Netcool server is running and the IP address matches the value in the **Host IP** field.

11. Right-click the form header and click **Save**.

12. Click **Test connector** to verify the connection between the MID Server and the Netcool connector.

13. If the test fails, follow the instructions that are issued by the error to correct the problem and then run another test.

**Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the Netcool device.

14. After a successful test, select the **Active** check box and then click **Update**.

Configure event collection from an Icinga2 connector

Configure the Icinga 2 (Icinga) connector instance to receive events while monitoring your network resources.

Supported version: 2.4.1

Role required: evt_mgmt_admin

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances**.

2. Click **New**.

3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th><strong>Connector Instance form</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>Descriptive and unique name for the Icinga connector.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Description for the use of the Icinga event collection instance.</td>
</tr>
<tr>
<td><strong>Connector definition</strong></td>
<td>Name of the required connector definition. Select <strong>Icinga2</strong>.</td>
</tr>
<tr>
<td><strong>Host IP</strong></td>
<td>IP address where Icinga is installed.</td>
</tr>
<tr>
<td><strong>Credential</strong></td>
<td>Permission to connect to Icinga. Click the search icon in the <strong>Credential</strong> field. Use Basic Authentication credentials. Either select the required credentials from the list or click <strong>New</strong> and create the required credentials on the Credentials form. If you create the credentials, save them using a unique and recognizable name, for example, <strong>ICINGA2CHCK</strong>.</td>
</tr>
<tr>
<td><strong>Event collection last run time</strong></td>
<td>Last run time value. Automatically updated.</td>
</tr>
<tr>
<td><strong>Last event collection status</strong></td>
<td>Last run status. Automatically updated.</td>
</tr>
<tr>
<td><strong>Event collection schedule (seconds)</strong></td>
<td>Frequency, in seconds, that the system checks for new events from Icinga. The default value is 120 seconds.</td>
</tr>
<tr>
<td><strong>Last error message</strong></td>
<td>Last error message. Automatically updated.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select **Save**.

The connector instance values are added to the form and the parameters that are relevant to the connector appear.
5. In the Connector Instance Values section, verify and where required, modify the default connector instance values.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>debug</td>
<td>Display debug messages. Default value: false. Specify true to see debug messages.</td>
</tr>
<tr>
<td>port</td>
<td>Number of the connector port. Default value: 5665.</td>
</tr>
<tr>
<td>protocol</td>
<td>Type of protocol. Default protocol type: https.</td>
</tr>
</tbody>
</table>

6. Optional: In the MID Servers for Connectors section, specify a MID Server that is up and is valid.

   If no MID Server is specified, an available MID Server that has a matching IP range is used.

   You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.

7. Right-click the form header and select **Save**.

8. Test the connection between the MID Server and the Icinga connector.
   a) Click **Test Connector**.
      
      If the test fails, follow the instructions that the error issues to correct the problem and then run another test.

      **Note**: Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the external monitoring tool.

   b) After a successful test, select the **Active** check box.

9. Click **Update**.

Configure event collection from NNMi

Configure the HP Network Node Manager i (NNMi) connector instance to receive events while monitoring your network resources.

Supported version: 10.30.653
Role required: evt_mgmt_admin

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click New.
3. On the form, fill in the fields.

**Connector Instance form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive and unique name for the NNMi connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the NNMi event collection instance.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Name of the required connector definition. Select NNMi.</td>
</tr>
<tr>
<td>Host IP</td>
<td>IP address where NNMi is installed.</td>
</tr>
<tr>
<td>Credential</td>
<td>Permission to connect to NNMi. Click the search icon in the Credential field. Either select the required credentials from the list or click New and create the required credentials on the Credentials form. If you create the credentials, save them using a unique and recognizable name, for example, NNMICHCK.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>Last run time value. Automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>Last run status. Automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>Frequency, in seconds, that the system checks for new events from NNMi. The default value is 120 seconds.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message. Automatically updated.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select Save.
   The connector instance values are added to the form and the parameters that are relevant to the connector appear.
5. In the Connector Instance Values section, verify and where required, modify the default connector instance values.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>days_from</td>
<td>Number of days for which events must be collected at the first collection cycle. Default value: 7 days.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The maximum number of past events that can be collected is 3,000. The first time that the connector runs, it collects all events (up to the 3,000 limit) that were created from the days_from date until the current date.</td>
</tr>
<tr>
<td>debug</td>
<td>Display debug messages. Default value: \texttt{false}. Specify \texttt{true} to see debug messages.</td>
</tr>
<tr>
<td>port</td>
<td>Number of the connector port. Default value: 80.</td>
</tr>
<tr>
<td>protocol</td>
<td>Type of protocol. Default protocol type: \texttt{http}.</td>
</tr>
</tbody>
</table>

6. Optional: In the MID Servers for Connectors section, specify a MID Server that is up and is valid.

   If no MID Server is specified, an available MID Server that has a matching IP range is used.

   You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.

7. Right-click the form header and select \textbf{Save}.

8. Test the connection between the MID Server and the NNMi connector.
   a) Click \textbf{Test Connector}.
      
      If the test fails, follow the instructions that the error issues to correct the problem and then run another test.
      
      **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the external monitoring tool.
   
   b) After a successful test, select the \textbf{Active} check box.

9. Click \textbf{Update}.

Configure event collection from NagiosXI

Configure the NagiosXI connector instance to receive events from the Nagios Core monitor.

Supported version: 5.5.2

The NagiosXI connector instance requires a credential that lets the instance access NagiosXI accounts. You can use an existing credential or \textbf{create a new one}.

Role required: \texttt{evt_mgmt_admin}

1. Navigate to \textbf{Event Management > Event Connectors (Pull) > Connector Instances}.
2. Click \textbf{New}.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the Nagios connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the NagiosXI event collection instance.</td>
</tr>
</tbody>
</table>
Field          | Description                                                                                       |
-------------|--------------------------------------------------------------------------------------------------|
Host IP      | Specify the Nagios XI IP address.                                                                |
Credential   | Select the credential with basic authentication that you created for this connector. For more information, see Create Nagios XI server credentials. Ensure that the user password contains the NagiosXI user API key, for example 04lquEPqf4JimWCm8RWBshOykOpW8L8YBIFeV4p09O05H5Ye4QDrHFPndYbW. |
Event collection last run time | The last run time value is automatically updated.                                                |
Last event collection status | The last run time status is automatically updated.                                            |
Event collection schedule (seconds) | The frequency, in seconds, that the system checks for new events from Nagios. The default value is 120 seconds. |
Last error message | The last error message is automatically updated.                                                |
Connector Definition | Select NagiosXI.                                                                                     |
MID Servers (MID Server for Connectors section) | Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section. |

4. Right-click the form header and select Save. The connector instance values are added to the form and the parameters that are relevant to the connector appear.

5. In the Connector Instance Values section, specify the Nagios values.
   a) initial_sync_in_days. Specify the number of days for which events must be collected at the first collection cycle. Default value 7 days.
   b) port. Default value 80.
   c) protocol. The default protocol type is http.
   d) time_zone. The default time zone is GMT.

6. Right-click the form header and select Save.
7. Click Test Connector to verify the connection between the MID Server and the connector.
8. If the test fails, follow the instructions that the error issues to correct the problem and then run another test.

   If this message appears: Connection test failed: Failed to connect to Nagios on test connector. Invalid API Key, then enter the API Key for the specific user to be able to read the Nagios events.

   Note: Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the Nagios Core monitor.

9. After a successful test, select the Active check box and then click Update.

Create Nagios XI server credentials
Create credentials to access Nagios XI server.
Role required: evt_mgmt_admin

1. Navigate to Connections & Credentials > Credentials.
2. Click New.
3. Select Basic Auth Credential.
4. On the form, fill in the fields.

### Basic Auth Credentials form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this credential. For example, you might call it Nagios authentication.</td>
</tr>
<tr>
<td>User name</td>
<td>Name to identify the user. Ensure that the user password contains the NagiosXI user API key, for example, 041quEPqf4JimWCm8RWbJokOpW8LYBUfEvJp90SHSRY.</td>
</tr>
<tr>
<td>Password</td>
<td>Password to use this credential.</td>
</tr>
<tr>
<td>Active</td>
<td>Option to enable the use of this credential.</td>
</tr>
<tr>
<td>Order</td>
<td>The order (sequence) in which the platform tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order. Default value: 100</td>
</tr>
</tbody>
</table>

5. Click Submit.

The credential for use with the Nagios XI server connector is created.

---

Configure OP5 connector

Configure the OP5 Monitor connector instance to receive alerts from an OP5 Monitor source.

Role required: evt_mgmt_admin

Supported version: 7.3.15

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click New and create a connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the Op5 connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the Op5 event collection instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the Op5 IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Either select the required credentials from the list or click New and create the required credentials. If you create the credentials, save them using a unique and recognizable name, for example, Op5OPS.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from Op5.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Select Op5.</td>
</tr>
<tr>
<td>Connector Instance Values section</td>
<td>The parameters that are specific to Op5 display here when the form is saved.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select Save. The connector instance values are added to the form and the parameters that are relevant to the connector appear.

4. In the Connector Instance Values section, specify the Op5 values.
   a) days_from. Specify the number of days for which events must be collected at the first collection cycle. Default value 7 days.
   b) debug. Default value false.
   c) port. Default value 443.
   d) state_types. Specify filters for event collection based on the state_types attribute. You can filter events based on state type. Specify the type, for example, soft or hard.

5. Right-click the form header and select Save.

6. Click Test Connector to verify the connection between the MID Server and the connector.

7. If the test fails, follow the instructions that the error issues to correct the problem and then run another test.

   If this message appears: Connection test failed: Failed to connect to Op5 on test connector. Invalid API Key, then enter the API Key for the specific user to be able to read the Op5 events.

   **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the OP5 Monitor source.

8. After a successful test, select the Active check box and then click Update.

Configure Opsview connector

Configure the Opsview Monitor connector instance to receive alerts from an Opsview Monitor source.

Role required: evt_mgmt_admin

Supported version: 5.4.0

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.

2. Click New and create a connector instance with the following details:
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the Opsview connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the Opsview event collection instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the Opsview IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Either select the required credentials from the list or click <strong>New</strong> and create the required credentials. If you create the credentials, save them using a unique and recognizable name, for example, OpsviewOPS.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from Opsview.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Select <strong>Opsview</strong>.</td>
</tr>
<tr>
<td>Connector Instance Values section</td>
<td>The parameters that are specific to <strong>Opsview</strong> display here when the form is saved.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select **Save**. The connector instance values are added to the form and the parameters that are relevant to the connector appear.

4. In the Connector Instance Values section, specify the Opsview values.
   a) **days_from**. Specify the number of days for which events must be collected at the first collection cycle. Default value 7 days.
   b) **debug**. Default value false.
   c) **eventtype_filter**. Specify filters for event collection based on the **eventtype** attribute. Specify one of these values 0,1,2,3
   d) **port**. Default value 80.
   e) **protocol**. The default protocol type is HTTPS.

5. Right-click the form header and select **Save**.
6. Click **Test Connector** to verify the connection between the MID Server and the connector.
7. If the test fails, follow the instructions that the error issues to correct the problem and then run another test.
   If this message appears: **Connection test failed: Failed to connect to Opsview on test connector. Invalid API Key**, then enter the API Key for the specific user to be able to read the Opsview events.
8. After a successful test, select the **Active** check box and then click **Update**.

Configure PRTG connector

Configure the PRTG connector instance to receive alerts from a Paessler PRTG Network Monitor source.

Role required: evt_mgmt_admin

Supported version: 18.2.4

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances.**
2. Click **New** and create a connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the PRTG connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the PRTG event collection instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the PRTG IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Either select the required credentials from the list or click <strong>New</strong> and create the required credentials. If you create the credentials, save them using a unique and recognizable name, for example, <strong>PRTGOPS</strong>.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from PRTG.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Select <strong>PRTG</strong>.</td>
</tr>
<tr>
<td>Connector Instance Values section</td>
<td>The parameters that are specific to <strong>PRTG</strong> display here when the form is saved.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select **Save**.

The connector instance values are added to the form and the parameters that are relevant to the connector appear.

4. In the Connector Instance Values section, specify the PRTG values.

   a) **add_columns**. Specify which columns to be returned by the connector. Separate the names with a comma and without a space between the names. The output is stored in the **Additional information** field of the event. Default value **false**. If **false** is specified, only the default list of columns is retrieved: parent, name, type, objid, tags, message, status, priority, datetime, actions, baselink, basetype, modifiedby.
**Note:** Only column names that are applicable to the **Message** content type can be specified.

b) **date_format.** The format date and time. Default format M/d/yyyy/ h:mm:ss a
   If you receive an event whose date is in a different format, modify this value to match the format of the incoming event. If you do not, the event will not be processed correctly.
   For example, if an event arrives on June 27, 2019 at 11:25 AM with a listed date of 2019/06/27/ 11:25:00 a, modify the **date_format** value to yyyy/M/d/ h:mm:ss a to match the format of the received event.
   In the **date_format**, a represents AM, and p represents PM.

c) **days_from.** Specify the number of days for which events must be collected at the first collection cycle. Default value 7 days.

d) **debug.** Default value false.

e) **port.** Default value 80.

f) **protocol.** The default protocol type is **http.**

g) **prtg_api_fetched_events_num.** Maximum number of events, per query, that can be fetched from the PRTG client into the MID Server. Default number: 50,000.

5. Right-click the form header and select **Save.**

6. Click **Test Connector** to verify the connection between the MID Server and the connector.

7. If the test fails, follow the instructions that the error issues to correct the problem and then run another test.
   If this message appears: **Connection test failed: Failed to connect to PRTG on test connector. Invalid API Key,** then enter the API Key for the specific user to be able to read the PRTG events.

**Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to PRTG.

8. After a successful test, select the **Active** check box and then click **Update.**

**Configure SAP Solution Manager connector**

Configure the SAP Solution Manager (Solman) connector instance to enable communication between the SAP Solution Manager and Event Management. This configuration enables you to receive events while monitoring your network resources.

Role required: admin

Supported versions: Tested on SAP Solution Manager 7.1 and 7.2. Might also work on other versions.

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances.**
2. Click **New** and create a connector instance.
3. Fill in the fields on the form.

**Connector instance form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the SAP Solman connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the SAP Solman event collection instance.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Active</td>
<td>Option that activates the connector instance. Select this option only after running a successful test.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select SAP Solman.</td>
</tr>
<tr>
<td>Host IP</td>
<td>The IP address where SAP Solman is installed.</td>
</tr>
<tr>
<td>Credential</td>
<td>Required credentials entered either by selecting them from the results of a search through the search icon or by clicking New, or creating them. If you create credentials, save them using a unique and recognizable name, such as SAPSOLMANOPS.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value. Automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last run time status. Automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from SAP Solman.</td>
</tr>
<tr>
<td>Bi-directional</td>
<td>Option to invoke the bi-directional option. This option enables the bi-directional exchange of values to-and-from the external event source. The Last bi-directional status option appears only when selecting this option.</td>
</tr>
<tr>
<td>Last bi-directional status</td>
<td>This field appears only when Bi-directional is selected. This field is automatically populated with the most recent bi-directional status.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message. Automatically updated.</td>
</tr>
<tr>
<td>Connector Instance Values section</td>
<td>The parameters that are specific to SAP Solman, automatically displayed when saving the form.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>MID Server that is up and valid. You can configure several MID Servers so that if the first server is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order in which their details are entered into the MID Server for Connectors section. If no MID Server is specified, an available MID Server that has a matching IP range is used.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select Save.

   The connector instance values are added to the form and the parameters that are relevant to the connector appear.

5. In the Connector Instance Values section, specify the default connector instance values.
   - **Client**: Default value = 001
   - **Port**: Default value = 8000

6. Right-click the form header and select Save.

7. Click Test Connector to verify the connection between the MID Server and the connector.

   If the test fails, follow the instructions in the error message and then run another test.

   **Note:** Use a network tool such as ping to verify the credentials and the network connectivity from the MID Server to the SAP Solman monitor.
8. After a successful test, select the **Active** check box and click **Update**.

SAP configurations enabling the SAP Solution Manager connector

Configure your SAP environment to work with the ServiceNow Event Management platform so you can use the SAP Solution Manager connector.

When integrating your SAP environment with Event Management, the SAP Solution Manager sends events through a MID Server to Event Management, which generates an alert from the event.

After the MID Server processes events, you can perform the following actions in the SAP Solution Manager:

- Add comments
- Confirm alerts
- Set status to **Externally Managed**

The following diagram illustrates the flow of data between SAP and Event Management.
Data moves between SAP and Event Management using both Push and Pull interfaces.

- **Pull interface:**
  
The Connector Update Queues table (em_connector_update_queue) lists manual updates of alerts and sends them to the SAP Solution Manager.
  
The table’s business rule identifies each manual update of alerts in Event Management and updates the connector queue with these changes. By default, changes to all alert fields are tracked. For information on adjusting this setting, see the SAP Solution Manager documentation.
  
The Event Management - Queue connector processor scheduled job removes alert changes from the queue and sends them to the MID Server. By default, the system performs
this process in batches of 1,000 alerts. You can customize this batch size by setting the `evt_mgmt.max_update_source_records` property.

**Note:** You must define the Pull interface to enable bi-directional functionality in the SAP Solution Manager connector. For details, see the SAP Solution Manager documentation.

- **Push interface:** Because SAP can only push data to an external source, the integration uses the Push interface to store alerts sent from SAP to Event Management.

  SAP alerts create events in Event Management that use both event rules and CI identification rules to create Event Management alerts.

  After Event Management creates alerts, a business rule listens for a change to the alert. If it detects a change, Event Management updates the SAP Solution Manager alerts. For details, see [Create a custom pull bi-directional connector](#).

Use the SAP Solution Manager Pull connector

The SAP Solution Manager Pull connector sends information from Event Management to the SAP Solution Manager. The Pull connector sends responses received from a Push connector in a bi-directional environment.

Role required: `evt_mgmt_admin`

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances**.
2. Click **New**.
   
   The **Connector Instance New Record** page appears.
3. Enter values in the required fields, which are indicated by a red asterisk.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the connector.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Search for and select <strong>SAP Solution Manager</strong>.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select <strong>Basic auth</strong> for basic authentication.</td>
</tr>
<tr>
<td>Host IP</td>
<td>The SAP Solution Manager IP address.</td>
</tr>
</tbody>
</table>

   The Connector Update Queues table ([em_connector_update_queue](#)) lists manual updates of alerts and sends an event to the SAP Solution Manager, using the Pull interface.

**Note:** When adding bi-directional functionality for a Push connector, the Connector instance name must match the Source Instance field of the event generated by that source. The **Connector Instance** field value is a combination of `SID + _ + Mandant`.

For example, the following image displays an event generated by the instance source **PSM_001**. To enable sending information back, you would have to name your connector instance **PSM_001**.

4. Set the following values for the connector instance:
   - **Client:** Represents the client value in SAP, usually 001.
   - **Port:** The SAP Solution Manager WebService port. Default value = 8000.
The following table lists the actions you can perform in Event Management, and the resulting updates occurring in SAP Solution Manager.

### Event Management actions and SAP Solution Manager updates

<table>
<thead>
<tr>
<th>Action in Event Management</th>
<th>Result in SAP Solman</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close an alert.</td>
<td>Confirms an alert in the alert inbox. The alert is removed.</td>
</tr>
<tr>
<td>Select the <strong>Maintenance</strong> setting in an alert.</td>
<td>Adds a comment indicating that the alert is in Maintenance mode.</td>
</tr>
<tr>
<td>Clear the <strong>Maintenance</strong> setting in an alert.</td>
<td>Removes the Maintenance mode comment.</td>
</tr>
<tr>
<td>Select the <strong>Acknowledged</strong> setting in an alert.</td>
<td>Adds a comment indicating that the alert is in the Acknowledged mode.</td>
</tr>
<tr>
<td>Clear the <strong>Acknowledged</strong> setting in an alert.</td>
<td>Removes Acknowledged mode comment.</td>
</tr>
</tbody>
</table>

Note the following behaviors:

- The Update queue business rule in the Alert table identifies each manual update of the alert and updates the connector queue accordingly. By default, changes to all alert fields are tracked.
- The Event Management — Queue Connector Processor job removes alert changes from the queue and sends them to the MID Server. By default, Event Management performs this dequeue process in batches of 1,000 alerts. You can configure the batch size by setting the `evt_mgmt.max_update_source_records` property.

Use the SAP Solution Manager Push connector

The MID Server web service Event Collector enables you to collect alerts sent from the SAP Solution Manager through event stream notification capabilities. The interface runs both a Push and a Pull interface to interact directly with the SAP Solution Manager.

- Deploy and start the MID Server.
- Configure and start the MID Web Server.
- Configure and start the MID Server web service Event Collector.

Role required: `evt_mgmt_admin`

SAP Solution Manager sends JSON formatted event messages to the MID Server. The MID Server parses the collected alert messages and transforms them using the SAP Solman Transform (`TransformEvents_SAPSolman`) script.

JSON formatted messages are collected using basic authentication. For details on other supported authentication methods, see [Configure the MID Web Server extension](#).

1. Navigate to **Event Management** > **Event Listener (Push)** > **Listener Transform Scripts**.
2. Select the `TransformEvents_SAPSolman` script.
3. Use the following URL to push alert messages from SAP Solution Manager to the MID Server:

   ```
   http://<MID_Server_IP>:<MID_Web_Server_Port>/api/mid/em/inbound_event?
   Transform=TransformEvents_SAPSolman, where the following are true:
   - **MID_Server_IP** is the IP address of the MID Web Server extension
   - **MID_Web_Server_Port** is the listening port of the MID Web Server extension
   ```

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Create an Remote Function Call (RFC) in the SAP Solution Manager and install a transport, as described in Configure RFC in SAP Solution Manager.

SAP Solution Manager setup configurations
As part of the SAP Solution Manager setup, you must perform several configurations to enable SAP Solution Manager to interact with Event Management.

Configure RFC in SAP Solution Manager
As part of enabling communication with Event Management, you must create an Remote Function Call (RFC) in the SAP Solution Manager and install a transport.

Download the following .psm files and import them into the SAP Solution Manager system:
- Transport File 1
- Transport File 2

Role required: evt_mgr_admin

1. On the SAP UI, enter the transaction code SM59 and create a new RFC with the following parameters on the Technical Settings tab:

<table>
<thead>
<tr>
<th>SAP Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Host</td>
<td>Name of the MID Server.</td>
</tr>
<tr>
<td>Service No</td>
<td>Listener port on the MID Server</td>
</tr>
<tr>
<td>Path Prefix</td>
<td>The URL of the MID Server transform script: /api/mid/em/inbound_event? Transform=TransformEvents_SAPSolman</td>
</tr>
</tbody>
</table>
2. On the **Logon & Security** tab, select **Basic Authentication** and enter your Event Management logon credentials in the **User** and **Password** fields.
A custom SAP BADI (Business Add-in) named Z_ALRTREACTION_IMPL is created and stored as a workbench transport request. After installation, this BADI calls the RFC to connect to Event Management.
Configure SAP to forward alerts to your BADI

After setting up SAP Solution Manager monitoring, you must configure SAP to forward alerts to the BADI (Business Add-in) you create as the first step in the SAP integration setup.

Role required: evt_mgt_admin

Third-party alerts get forwarded globally. You can configure the alert reaction only for a specific monitoring template.

1. On the SAP UI, navigate to Application Operations > System Monitoring.
2. In the ribbon at the top of the System Monitoring configuration page, select Configure Infrastructure > 2.3 Default Settings.
   The Default Settings page appears.
3. Configure and activate a third-party component.
   a) Enter edit mode by clicking the **Edit** button.
   b) Select the **Third-Party Components** tab.
   c) Click **Add** and add the following value as the third-party connector: **Implementation:** BADI Definition for Alert Reactions ZSNOW with scope = All Alerts and Metrics.
   d) Click **Save** to activate third-party components globally.

4. Define the objects to which the monitoring template will be assigned.
   a) In the ribbon at the top of the page, select **5 Define Scope**. The Define Scope page appears.
b) Select the managed object to which you want to assign the monitoring template. The monitoring template is assigned to the selected objects.

5. Reactivate the monitoring of templates.
   a) Select 6 Setup Monitoring.
   b) Search for and locate entries with Setup Status = Reconfigure Required and assign the monitoring template to those entries.
   c) Select Apply and Activate. When the jobs finish, verify that the Assignment Status is green. All templates and active alerts assigned to the selected objects use the new forwarding method.

View alerts generated in SAP Solution Manager to see any pressing issues, as described in View alerts in the SAP Solution Manager inbox.

View alerts in the SAP Solution Manager inbox
You can view alerts generated in SAP Solution Manager to see any pressing issues. All alerts also forward to Event Management automatically.

Role required: evt_mgt_admin

1. On the SAP UI, open Solution Manager and select the System Alerts tab.
The system alerts are listed on the page.

2. Navigate to (Event Management > All Alerts) to view the alerts in Event Management.
The following table describes the field mapping between SAP Solution Manager and Event Management for alerts.

**Field mapping for SAP Solution Manager and Event Management**

<table>
<thead>
<tr>
<th>SAP Field</th>
<th>Event Management Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>severity</td>
<td>severity</td>
<td>SAP uses a number from 1 (less critical) to 9 (more critical)</td>
</tr>
<tr>
<td>message_key</td>
<td>message_key</td>
<td>SAP message group ID, messageGUID</td>
</tr>
<tr>
<td>description</td>
<td>description</td>
<td>Event description</td>
</tr>
<tr>
<td>source</td>
<td>source</td>
<td>SAP system ID</td>
</tr>
<tr>
<td>managed_object_type</td>
<td>type</td>
<td>The type of the object, such as Host or Instance</td>
</tr>
<tr>
<td>managed_object_name</td>
<td>node</td>
<td>Object name</td>
</tr>
<tr>
<td>node</td>
<td>resource</td>
<td>Solution manager node name</td>
</tr>
<tr>
<td>N/A (there is no value provided by SAP, Event Management generates the indicated field.)</td>
<td>source instance</td>
<td>The event class must match the connector pull instance name, with the structure &lt;Source&gt;_&lt;Client&gt;.</td>
</tr>
<tr>
<td>source</td>
<td>additional_info.sid</td>
<td>SAP system ID</td>
</tr>
<tr>
<td>rating</td>
<td>additional_info.rating</td>
<td>Event color in SAP</td>
</tr>
<tr>
<td>resource</td>
<td>additional_info.mandant</td>
<td>SAP mandant</td>
</tr>
</tbody>
</table>
View and update your SAP Solution Manager certificate
View your SAP Solution Manager certificate, and update the certificate if necessary.

Role required: evt_mgmt_admin

1. Connect to SAP on client 00.
2. Enter the \texttt{strust} transaction.
3. In the Trust Manager UI, open the SSL client.
4. Import the SSL certificate and save it.
5. Use the \texttt{smicm} transaction to enable the system to work with both HTTP and HTTPS clients.

The ICM module restarts.

View the SAP interface log
View the SAP interface log to help you debug issues.
Role required: admin

Select a transaction in the SAP UI.
The ZSNOW interface debugging table appears.

SAP Solution Manager transaction codes
The transaction code abbreviations that you can use in the SAP interface when working with the SAP Solution Manager connector.

<table>
<thead>
<tr>
<th>Transaction Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALRTINBOX</td>
<td>Displays the alert inbox.</td>
</tr>
<tr>
<td>SOAMANGER</td>
<td>Opens and activates the <a href="https://example.com">Sap_Solman_Alert_Inbox</a> web service.</td>
</tr>
<tr>
<td>SRT_UTIL</td>
<td>View the SOA manager logs.</td>
</tr>
<tr>
<td>SOLMAN_WORKCENTER</td>
<td>Opens the SAP Solution Manager interface in a browser.</td>
</tr>
<tr>
<td>SM59</td>
<td>Displays the entire output interface.</td>
</tr>
<tr>
<td>SE38</td>
<td>Executes the Advanced Business Application Programming (ABAP) code. (The equivalent ServiceNow code is ZTEST_SNOWCONNECTOR.)</td>
</tr>
</tbody>
</table>
Configure alert collection from SCOM
Alerts from the Microsoft System Center Operations Manager (SCOM) are collected using the SCOM connector instance.

The SCOM connector instance requires configuration before receiving alerts and Operational Intelligence raw data from the SCOM server. You can limit the collection of SCOM alerts to only those alerts that belong to the SCOM group that you specify.

Configure the SCOM connector instance
Configure the Microsoft System Center Operations Manager (SCOM) connector instance to receive alerts and Operational Intelligence raw data from the SCOM server.

Role required: evt_mgmt_admin

Supported versions:
- 2007 – version 6.1.7221.0
- 2012 - version 7.1.10226.0
- 2016 - version 7.2.117190 and 7.3.13261.0

Before starting the procedure to configure the connector, ensure:
- The MID Server is running with a user that has local admin permissions to enable the MID Server to run PowerShell and the user has read access to the SCOM API.
- The MID Server resides in the same domain as the SCOM server.
- The MID Server uses the same time zone as the SCOM server.
- The MID Server is running on Windows.
- The MID Server has .NET framework version 3.5.
- If you want to activate Metric Collection, ensure that the MID Server that retrieves metrics is configured with the Operational Intelligence extension and that the extension is in Started mode. See Manually configure the Operational Intelligence extension.

If Bi-directional is selected, ensure that:
- PowerShell version 3.0 is installed on Windows.
- That the MID Server is running with a user that has local admin permissions to enable the MID Server to run PowerShell.

If Metric Collection is selected and the Database login with Windows authentication option is also selected, ensure that:
- As Windows authentication is used by the connector to access the SCOM database (OperationsManagerDW), the MID Server service should be running with a user having read access to the SCOM database. Ensure that the correct credentials are used:
  1. In the local services, right-click the MID Server service and select Properties.
  2. In the Log On tab, ensure that This account is selected with the details of the user in the Windows domain having read access to the SCOM database.
Note: If the OperationsManagerDW database has been renamed, also change the database name in the SCOMConnector.groovy MID Server Script as well as in the MetricCollector script include.

- If you upgraded from a release earlier than Istanbul and a SCOM connector was defined:
  1. Define the Log On user on the MID Server service.
  2. Ensure that This account is selected with the details of the user in the Windows domain having read access to the SCOM database.

To receive SCOM alerts, you can obtain the redistributable SCOM files from your SCOM application. Add the files to the MID Server, and then configure a SCOM connector instance to collect the alerts and Operational Intelligence raw data.

The collection of Operational Intelligence is performed using JavaScript running JDBC in place of PowerShell.

1. On the SCOM server, download the following files to a local computer.

<table>
<thead>
<tr>
<th>Version and location</th>
<th>SCOM path and library names</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCOM 2012R2 or SCOM 2012</td>
<td>%ProgramFiles%\Microsoft System Center 2012 R2 or 2012\Operations Manager\Server\SDK Binaries</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.Core.dll</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.OperationsManager.dll</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.Runtime.dll</td>
</tr>
<tr>
<td>SCOM 2007</td>
<td>%ProgramFiles%\System Center Operations Manager 2007\SDK Binaries</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.OperationsManager.dll</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.Runtime.dll</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Version and location</th>
<th>SCOM path and library names</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCOM 2016</td>
<td>%ProgramFiles%\Microsoft System Center 2016\Operations Manager \Server\SDK Binaries</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.Core.dll</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.OperationsManager.dll</td>
</tr>
<tr>
<td></td>
<td>• Microsoft.EnterpriseManagement.Runtime.dll</td>
</tr>
</tbody>
</table>

Note: The MID Server must be installed with .NET 4.6 or higher.

Note: A connection cannot be made to SCOM 2012 and SCOM 2007 from the same MID Server. To work with both SCOM 2012 and SCOM 2007 in your instance, before uploading the following files to your instance, append .2012 to the end of the filename Microsoft.EnterpriseManagement.OperationsManager.dll that is found in the 2012 path and append .2007 to the end of the filename.
2. Navigate to MID Server > JAR Files.
3. Click **New** and add a separate record for the SCOM version for each of the DLL files that you downloaded from the SCOM server.
   a) In the **Name** field, specify the SCOM version and an identifier to make the name unique, for example 2012-1. If you are using SCOM 2016, specify 2012 as the version.
   b) Click the paper clip icon in the form header and then attach one of the appropriate DLL files that you downloaded.
   c) Click **Submit**.
4. Repeat step 3, creating a separate record for each of the remaining DLL files. Ensure that you have a unique identifier after the SCOM version for each file that you attach, for example 2012-2.
6. Click **New** and create a connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name for the SCOM connector instance.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the SCOM event collection instance.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this option only after running a successful test.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the <strong>SCOM</strong> connector definition.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the SCOM IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select Windows credentials.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last event collection run time value is automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last event collection run time status is automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from SCOM Operations.</td>
</tr>
<tr>
<td>Bi-directional</td>
<td>Select to invoke the bi-directional option. This option enables the bi-directional exchange of values to-and-from the external event source. There is default implementation for SCOM. The <strong>Last bi-directional status</strong> option displays only when this option is selected.</td>
</tr>
<tr>
<td>Last bi-directional status</td>
<td>The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message field is automatically updated. This message is the last error message received by the connector. If the test connector fails, an error message is displayed in this field.</td>
</tr>
<tr>
<td>Metrics collection</td>
<td>Select to enable the collection of Operational Intelligence.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Metrics collection last run time</td>
<td>The last run time of the metrics collection scheduled job. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Last metrics collection status</td>
<td>Status of the Operational Intelligence collection activity. The value of this field is automatically populated. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics collection schedule (seconds)</td>
<td>The time, in seconds, to repeat the Operational Intelligence collection scheduled job. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics database host</td>
<td>The IP address or the host name of the metrics database host. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Connect using a named instance</td>
<td>When selected, the connection is made using the specified named instance. Otherwise, the connection is made using the specified port. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics database port</td>
<td>The port used by the metrics database. The connection is made using JDBC. Default port number 1,433. This option displays only when the <strong>Metrics collection</strong> option is selected. However, if <strong>Connect using a named instance</strong> is selected, this option no longer displays.</td>
</tr>
<tr>
<td>Metrics database named instance</td>
<td>The metrics database instance name. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Database login with Windows authentication</td>
<td>Perform database login with the credentials of the log-on user that is defined on the MID Server service. This option displays only when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics database credential</td>
<td>Credentials for the metric database. Use JDBC credentials for the local database user. This option displays only when the <strong>Metrics collection</strong> option is selected. However, if <strong>Database login with Windows authentication</strong> is selected, this option no longer displays.</td>
</tr>
</tbody>
</table>
### Field | Value
--- | ---
MID Servers (MID Server for Connectors section) | Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section. The port requirement from the MID Server to the SCOM server is 5724 and this is fixed. If the specified MID Server is in a cluster, it might not be selected and another available MID Server is then selected in its place.

7. Right-click the form header and select **Save**.
   The connector instance values are added to the form and the parameters that are relevant to the connector appear.

8. In the Connector Instance Values section, you can edit the values of the mandatory SCOM parameters.
   a) **login_with_windows_authentication** Default value: **false**.
      Set to **true** to enable SCOM event collection and the bi-directional exchange of event values to work with Windows authentication.
      When invoking this value, ensure that you do the following on the MID Server:
      1. Navigate to the list of local services, right-click the MID Server service and select **Properties**.
      2. In the **Log On** tab, ensure that **This account** is selected with the details of the user in the Windows domain having read access to the SCOM database.

   b) **metric_chunk_size** Default value 50000.

   c) **scom_date_format** Default format M/d/yyyy/ h:mm:ss a
      If you receive an event whose date is in a different format, modify this value to match the format of the incoming event. If you do not, the event will not be processed correctly.
      For example, if an event arrives on June 27, 2019 at 11:25 AM with a listed date of 2019/06/27/ 11:25:00 a, modify the **scom_date_format** value to yyyy/M/d/ h:mm:ss a to match the format of the received event.
      In **scom_date_format**, a represents AM, and p represents PM.

   d) **scom_initial_sync_in_days** Default value 7.

   e) **scom_version** It is mandatory to specify the SCOM version, select from 2016, 2012, or 2007.

9. Right-click the form header and select **Save**.
10. Restart the MID Server service to copy the files.
11. Click **Test connector** to verify the connection between the MID Server and the connector.
12. If the test fails, correct the problem by following the instructions issued by the error and then run another test.

   **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to SCOM.

13. After a successful test, select the **Active** check box and then click **Update**.
Note:
The default binding rules that contain SCOM as the external source, that applies to IT alerts and Operational Intelligence raw data, are the following SCOM Management Packs:

- All OS Management Packs
- MS SQL Server
- IIS

If bi-directional is configured, the bi-directional exchange of values to-and-from the external event source is enabled.

These scenarios describe the default bi-directional functionality for SCOM connectors:

- When an alert is resolved in SCOM, it is auto-closed in ServiceNow. However, it is updated irrespective of the bi-directional feature because during each collection cycle, all alert changes are updated.
- When an alert is manually closed in ServiceNow, it is auto-closed in SCOM. If the alert state is changed to Reopen, SCOM is also updated.
- When an incident is created and associated to an alert in ServiceNow, SCOM receives the incident number as a ticket ID. However, the state of the incident is not available on SCOM. Therefore when the incident is resolved in ServiceNow, SCOM is not updated as the incident number remains the same. When the alert is associated with a new incident, the new incident number is updated in SCOM.

Limit collected SCOM alerts to specific SCOM groups
Limit the collection of SCOM alerts to only those alerts that belong to the specified SCOM group.

1. Use the SCOM Authoring workspace and Operations console to define a SCOM group. Ensure that the group contains the required computers or instances. For further information about SCOM groups, see the SCOM documentation.
2. Create a role that has a scope of the SCOM group.
3. Add the new role to the SCOM user.
4. Assign the role to the user defined in the credentials given to the SCOM connector instance.
5. Remove all other roles from the user.

Only alerts from the specified SCOM group arrive at the instance.

Configure event collection from SolarWinds monitor
Configure the SolarWinds monitor connector instance to receive events from the SolarWinds monitor.

Supported versions:

- SAM 6.2.1
- NPM 12.0

Note: The SolarWinds monitor connector instance can import events from:

- Network Performance Monitor (NPM)
- SolarWinds Service & Application Monitor (SAM)

The SolarWinds monitor connector instance requires a credential that lets the instance access SolarWinds monitor accounts. You can use an existing credential or create a new one.

Role required: evt_mgmt_admin and evt_mgmt_integration

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click New.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Type a descriptive name for the SolarWinds monitor connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for the use of the SolarWinds monitor event collection instance.</td>
</tr>
<tr>
<td>Connector Definition</td>
<td>Select Solarwinds.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the SolarWinds monitor IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select the credential with basic authentication that you created for this connector. For more information, see Create SolarWinds monitor credentials.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>The last run time value of the scheduled job. This field is updated automatically.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>The last event collection status. This field is updated automatically.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from the SolarWinds monitor server.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message received by the connector is automatically updated.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select Save.
The connector instance values are added to the form and the parameters that are relevant to the connector appear.

5. In the Connector Instance Values section, specify these fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial_sync_in_days</td>
<td>Specify how many days the initial pull must contain to retrieve events. The default value is 7 days.</td>
</tr>
<tr>
<td>nodes_custom_properties</td>
<td>Specify the SolarWinds monitor property names that are associated with the relevant node. This information is added to the Additional Information field of the event in JSON format. The SolarWinds monitor custom properties are user-defined fields, for example, country, building, or serial number, that you can associate with monitored network objects. Separate multiple entries with a comma.</td>
</tr>
</tbody>
</table>
Create SolarWinds monitor credentials

Create credentials to access SolarWinds monitor.

Role required: evt_mgmt_admin

1. Navigate to Connections & Credentials > Credentials.
2. Click New.
3. Select Basic Auth Credential.
4. On the form, fill in the fields.

### Basic Auth Credentials form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this credential. For example, you might call it SolarWinds authentication.</td>
</tr>
<tr>
<td>User name</td>
<td>Name to identify the user.</td>
</tr>
<tr>
<td>Password</td>
<td>Password to use this credential.</td>
</tr>
<tr>
<td>Active</td>
<td>Option to enable the use of this credential.</td>
</tr>
<tr>
<td>Order</td>
<td>The order (sequence) in which the platform tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order. Default value: 100</td>
</tr>
</tbody>
</table>

5. Click Submit.

The credential for use with the SolarWinds monitor connector is created.
Configure event collection from vCenter

Configure the VMware vCenter Server (vCenter) connector instance to receive events from your VMware vSphere environment.

Events collected by this connector are events that match the vCenter ESX Servers event rule.

Supported version: 6.5

Role required: evt_mgmt_admin

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click New.
3. On the form, fill in the fields.

**Connector Instance form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive and unique name for the vCenter connector.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the vCenter event collection instance.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Name of the required connector definition. Select vCenter.</td>
</tr>
<tr>
<td>Host IP</td>
<td>IP address where vCenter is installed.</td>
</tr>
<tr>
<td>Credential</td>
<td>Permission to connect to vCenter. Click the search icon in the Credential field. Either select the required credentials from the list or click New and create the required credentials on the Credentials form. If you create the credentials, save them using a unique and recognizable name, for example, vCenterCHCK.</td>
</tr>
<tr>
<td>Event collection last run time</td>
<td>Last run time value. Automatically updated.</td>
</tr>
<tr>
<td>Last event collection status</td>
<td>Last run status. Automatically updated.</td>
</tr>
<tr>
<td>Event collection schedule (seconds)</td>
<td>Frequency, in seconds, that the system checks for new events from vCenter. The default value is 120 seconds.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message. Automatically updated.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select Save.
The connector instance values are added to the form and the parameters that are relevant to the connector appear.

5. In the Connector Instance Values section, specify the vCenter values.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>collect_stateless_events</td>
<td>Events that have a status (stateful) are collected when this field is set to <strong>false</strong> (default). These events are of type AlarmStatusChangedEvent and AlarmClearedEvent. If this field is set to <strong>true</strong>, events of other types are collected as well.</td>
</tr>
<tr>
<td>port</td>
<td>Specify the value of the port. Default value: <strong>443</strong>.</td>
</tr>
<tr>
<td>protocol</td>
<td>Specify the protocol. Default value: <strong>https</strong>.</td>
</tr>
</tbody>
</table>

6. Optional: In the MID Servers for Connectors section, specify a MID Server that is up and is valid.
   If no MID Server is specified, an available MID Server that has a matching IP range is used.
   You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.

7. Right-click the form header and select **Save**.

8. Test the connection between the MID Server and the vCenter connector.
   a) Click **Test Connector**.
      
      If the test fails, follow the instructions that the error issues to correct the problem and then run another test.
      
      **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the external monitoring tool.

   b) After a successful test, select the **Active** check box.

9. Click **Update**.

Configure event collection from vRealize
Configure the VMware vRealize Operations (vRealize) connector instance to receive events from
the vRealize Operations Log and Event Management servers.

Supported version: 6.5.0

**Note:** Connect to vRealize using a local account with at least Read-only vRealize
Operations permission. You can use an existing credential or [create a new one](#).

Role required: evt_mgmt_admin

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click **New** and create a vRealize Operations connector instance with the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a unique name for the vRealize Operations connector instance.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Specify the vRealize Operations IP address.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select the credential with basic authentication that you created for this connector. For more information, see <a href="#">Create vRealize credentials</a>.</td>
</tr>
<tr>
<td>Schedule (seconds)</td>
<td>The frequency in seconds that the system checks for new events from vRealize Operations.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for the use of the vRealize Operations connector.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>The vendor and protocol used to gather events from the external event source. Select the vRealize connector definition.</td>
</tr>
<tr>
<td>Last error message</td>
<td>The last error message is automatically updated.</td>
</tr>
<tr>
<td>Last run time</td>
<td>The last run time value is automatically updated.</td>
</tr>
<tr>
<td>Last run status</td>
<td>The last run time status is automatically updated.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select **Save**.
   The connector instance values are added to the form and the parameters that are relevant
   to the connector appear.

4. In the Connector Parameters section, specify the values of the mandatory vRealize Operations parameters.
   a) **port** Default value 443.
b) **protocol** Default value https.

5. Click Test connector to verify the connection between the MID Server and the vRealize Operations connector.

6. If the test fails, follow the instructions that are issued by the error to correct the problem and then run another test.

   **Note:** Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to vRealize.

7. After a successful test, select **Active** and then click **Update**.

Create vRealize credentials

Create credentials to access vRealize.

Role required: evt_mgmt_admin

Create Basic Auth type of credential to connect to vRealize.

1. Navigate to **Connections & Credentials > Credentials**.
2. Click **New**.
3. Select **Basic Auth Credential**.
4. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this credential. For example, NewvRealize.</td>
</tr>
<tr>
<td>User name</td>
<td>Name to identify the user.</td>
</tr>
<tr>
<td></td>
<td>• If the username is internal in vRealize, use it as is.</td>
</tr>
<tr>
<td></td>
<td>• If vRealize is integrated with Active Directory (AD), use the format</td>
</tr>
<tr>
<td></td>
<td>DOMAIN\username@sourceauth. For example, local\administrator@All vCenter Servers.</td>
</tr>
<tr>
<td>Password</td>
<td>Password to use this credential.</td>
</tr>
<tr>
<td>Active</td>
<td>Option to enable the use of this credential.</td>
</tr>
<tr>
<td>Order</td>
<td>The order (sequence) in which the platform tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order. Default value: 100</td>
</tr>
</tbody>
</table>

5. Click **Submit**.

The credential for use with the vRealize connector is created.
Configure event collection from Zabbix server
Configure the Zabbix server connector instance to receiving alerts from the Zabbix server.

Supported version: 3.0.3

The Zabbix server connector instance requires a credential that lets the instance access Zabbix server accounts. You can use an existing credential or create a new one.

Role required: evt_mgmt_admin

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click New.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the connector.</td>
</tr>
<tr>
<td>Host IP</td>
<td>IP address for the Zabbix server.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select the credential with basic authentication that you created for this connector. For more information, see Create Zabbix server credentials.</td>
</tr>
<tr>
<td>Schedule (seconds)</td>
<td>Frequency, in seconds, that the system checks for new events from the Zabbix server.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the use of the Zabbix server event collection instance.</td>
</tr>
<tr>
<td>Connector Definition</td>
<td>Name of the connector definition. Select Zabbix.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message. Automatically updated.</td>
</tr>
<tr>
<td>Last run time</td>
<td>Last run time value. Automatically updated.</td>
</tr>
<tr>
<td>Last run status</td>
<td>Last run time status. Automatically updated.</td>
</tr>
<tr>
<td>MID Servers (MID Server for Connectors section)</td>
<td>Optional. Name of a MID Server. If no MID Server is specified, an available MID Server that has a matching IP range is used. In the MID Servers for Connectors section, specify a MID Server that is up and valid. You can configure several MID Servers. If the first is down, the next MID Server is used. If that MID Server is not available, the next is selected, and so on. MID Servers are sorted according to the order that their details were entered into the MID Server for Connectors section.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select Save.
The connector instance values are added to the form and the parameters that are relevant to the connector appear.

5. In the Connector Instance Values section, the default connector instance values appear. Verify and modify the values, as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>days_from</td>
<td>Number of days for which events must be collected at the first collection cycle. Default value: 2</td>
</tr>
<tr>
<td></td>
<td>Note: The maximum number of past events that can be collected is 3,000. The first time that this connector runs, the number of past events is counted in the past from the current date.</td>
</tr>
<tr>
<td>port</td>
<td>Port number. Default: 80</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol type. Default: https</td>
</tr>
</tbody>
</table>

6. Right-click the form header and select **Save**.
7. Click **Test Connector** to verify the connection between the MID Server and the connector.
8. If the test fails, follow the instructions that are issued by the error to correct the problem and then run another test.

Note: Use a network tool, such as ping, to verify credential correctness and network connectivity from the MID Server to the Zabbix server.

9. After a successful test, select **Active** and then click **Update**.

Create Zabbix server credentials
Create credentials to access Zabbix server.

Role required: evt_mgmt_admin

1. Navigate to **Connections & Credentials > Credentials**.
2. Click **New**.
3. Select **Basic Auth Credential**.
4. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for this credential. For example, you might call it Zabbix authentication.</td>
</tr>
<tr>
<td>User name</td>
<td>Name to identify the user.</td>
</tr>
<tr>
<td>Password</td>
<td>Password to use this credential.</td>
</tr>
<tr>
<td>Active</td>
<td>Option to enable the use of this credential.</td>
</tr>
<tr>
<td>Order</td>
<td>The order (sequence) in which the platform tries this credential as it attempts to log on to devices. The smaller the number, the higher in the list this credential appears. Establish credential order when using large numbers of credentials or when security locks out users after three failed login attempts. If all the credentials have the same order number (or none), the instance tries the credentials in a random order. Default value: 100</td>
</tr>
</tbody>
</table>

5. Click Submit.

The Basic Auth credential for use with the Zabbix server connector is created.

Create a custom pull connector

You can create a customized pull connector that requires a script, connector definition, and connector instance, to retrieve events on behalf of an event source.

Role required: evt_mgmt_admin

A custom connector script can make remote API calls for a new event source to send events to the ServiceNow instance. You can create a script, and then add it as part of a new Event Management connector definition and connector instance. The custom code that accomplishes these actions must:

- Connect to an event monitoring tool.
- Retrieve events from an event monitoring tool.
- Send events to the Event (em_event) table using a web service API. See .

1. Create a custom MID Server script include. This example uses Groovy (deprecated). In place of a script include in Groovy, it is recommended to use JavaScript.
   a) Implement these methods:
      - @Override OperationStatus testConnection()
      - @Override OperationStatus execute()
   b) Design the class to extend the ThirdPartyConnector.

```java
public class HypericConnector extends ThirdPartyConnector
```

c) Import platform classes for event creation, sending, logging, and third-party connector base classes.

```java
package com.service_now.mid.probe.tpcon.test
import com.glide.util.Log
```
import com.service_now.mid.MIDServer
import com.service_now.mid.probe.event.IEventSender
import com.service_now.mid.probe.event.SNEventSender
import com.service_now.mid.probe.tpcon.OperationStatus
import com.service_now.mid.probe.tpcon.OperationStatusType
import com.service_now.mid.probe.tpcon.ThirdPartyConnector
import com.snc.commons.eventmgmt.Event

2. Navigate to **MID Server** > **MID Server Script Files** and create a script. Specify the **Parent** field as **Groovy**, complete the form as appropriate, and then click **Submit**.

3. Navigate to **Event Management** > **Event Connectors (Pull)** > **Connector Definitions** and create a connector definition.

4. In the **Groovy script to run** field, select the MID Server script file, complete the form as appropriate. In addition to **username** or **host**, you can add any other parameter, for example, **port**, and then click **Submit**.

5. Navigate to **Event Management** > **Event Connectors (Pull)** > **Connector Instances** and create a connector instance.

6. In the **Connector definition** field, select the connector definition, complete the form as appropriate, and then click **Submit**.

7. To confirm or debug the script, use debug printouts from Groovy to the MID Server log.

8. To monitor incoming events using the custom connector instance, navigate to **ECC > Queue** and filter on **ConnectorProbe**.

Custom connector code sample

The code sample used in this connector definition demonstrates how to manage events using web service API.
Note: The sample connector code is not updated automatically. Check the connector code on the instance from time-to-time as it might be updated.

You can use the Find Packages Calls script to generate proposed changes to remove Packages calls or replace them with GlideScriptable names. For more information, see Packages Call Removal tool.

You can use the following code sample to create a JavaScript connector for receiving events from a SolarWinds external source.

```javascript
var SolarWindsJS = Class.create();

var SUCCESS = Packages.com.service_now.mid.probe.tpcon.OperationStatusType.SUCCESS;
var FAILURE = Packages.com.service_now.mid.probe.tpcon.OperationStatusType.FAILURE;
var Event = Packages.com.snc.commons.eventmgmt.Event;
var SNEventSenderProvider = Packages.com.service_now.mid.probe.event.SNEventSenderProvider;
var HTTPRequest = Packages.com.glide.communications.HTTPRequest;

var SOLAR_WINDS = "SolarWinds";
var MAX_EVENTS_TO_FETCH = 3000;
var errorMessage = "";

SolarWindsJS.prototype = Object.extendsObject(AProbe, {
    // test the connection with SolarWinds server
    testConnection : function() {
        ms.log("Solarwinds testing connection");

        var retVal = {};

        try {
            var response = this.getResponse(this.getQueryForTestConnection());
            if (response == null){
                retVal['status'] = FAILURE.toString();
                retVal['error_message'] = errorMessage;
                return retVal;
            }
            ms.log('SolarWindJS Connector Test Connection response:');
            if (response.getStatusCode() == 200){
                ms.log(response.getBody());
                retVal['status'] = SUCCESS.toString();
            }
            else{
                this.addError(response.getErrorBody());
                retVal['status'] = FAILURE.toString();
            }
            ms.log('result:' + response.getStatusCode());
            catch (e) {
                retVal['status'] = FAILURE.toString();
            }
            ms.log("SolarWindsJS Connector testConnection " + retVal['status'] );
            if (retVal['status'] === FAILURE.toString())
            retVal['error_message'] = errorMessage;
            return retVal;
        }
    }
});
```
execute: function() {
  ms.log("SolarWindJS Connector: execute connection ...");
  var retVal = {};

  var resultArray = this.getResult(this.getQueryForExecute()); // retrieve all events from SolarWinds
  var events = this.getSNEvents(resultArray); // convert raw events to SN events
  if (events == null) {
    retVal['status'] = FAILURE.toString();
    retVal['error_message'] = errorMessage;
    return retVal;
  }

  // send all events
  var sender = SNEventSenderProvider.getEventSender();
  var i = 0;
  var successFlag = true;
  for (; i < events.length; i++) {
    if (events[i]) {
      successFlag = successFlag && sender.sendEvent(events[i]); // send each event
    }
  }

  if (successFlag) {
    retVal['status'] = SUCCESS.toString();
    if (events.length > 0) {
      this.updateLastSignature(events, retVal); // if all events were sent successfully, update last signature
    } else {
      retVal['status'] = FAILURE.toString();
      retVal['error_message'] = errorMessage;
      return retVal;
    }
  }

  ms.log("SolarWindJS Connector: sent " + events.length + " events. Return to instance: status=" + retVal['status'] + " lastDiscoverySignature=" + retVal['last_event'] );

  return retVal;
},

getSNEvents: function(resultArray) {
  if (resultArray == null)
    return null;
  var events = [];

  // if no events were found, return
  if (resultArray.results.length == 0)
    return events;

  ms.log("resultArray.results.length: " + resultArray.results.length);

  // init all maps with additional information for events
  var eventTypes = this.getEventTypes();
var applications = this.getApplications();
var components = this.getComponents();
var nodes = this.getNodes();
var volumes = this.getVolumes();
var interfaces = this.getInterfaces();
var nodeProperties = this.getNodesProperties();

if (eventTypes == null || nodes == null) {
    return null;
}

var latestTimestamp = this.probe.getParameter("last_event");
var i = 0;
for (; i<resultArray.results.length; i++) {
    var event = this.createSNEvent(resultArray.results[i],
    eventTypes,
    applications, components, nodes, volumes, interfaces,
    nodeProperties);

    // filter out events on first pull
    if (!this.filterEvent(latestTimestamp, event)) {
        events.push(event);
    }
}

return events;

//ignore closed and info events on first action of pulling
filterEvent : function (latestTimestamp, event) {
    if (latestTimestamp == null){
        //checking if event is closed
        if( event.isClosing()){ 
            return true;
        }
        //checking if event is older than time period
        //time period format yyyy-MM-dd' 'HH:mm:ss.mmm
        var timeOfEvent = event.getTimeOfEvent().split(' ');
        var eventDate=timeOfEvent[0].split('-');
        var year= eventDate[0];
        var month= eventDate[1];
        var day=eventDate[2];
        //javascript month starts from 0
        var timeOfEventInMilis =new Date(year,month-1,day,0,0,0,0).getTime();
        var initialSyncDays=this.probe.getAdditionalParameter("initial_sync_in_days");

        //round to midnight
        var selectedTimePeriod= new Date().setHours(0,0,0,0)-(initialSyncDays*24*60*60*1000);

        if(selectedTimePeriod>timeOfEventInMilis){
            ms.log("event with time stamp " + event.getTimeOfEvent()+" will be filtered out. It is older than "+initialSyncDays +" last days");
            return true;
        }
    }
}

return false;
getQueryForTestConnection : function () {
    var query = "SELECT TOP 10 NodeID FROM Orion.Nodes";
    return query;
},

ggetQueryForExecute : function () {
    var latestTimestamp = this.probe.getParameter("last_event");
    var query = "SELECT TOP " + MAX_EVENTS_TO_FETCH + " EventID, EventTime, NetworkNode, NetObjectID, EventType,
    Message, Acknowledged, " +
    "NetObjectType, Timestamp FROM Orion.Events " +
    "WHERE NetworkNode > 0 AND NetObjectID > 0 ";
    if (latestTimestamp != null) { //differ between first collection
cycle to others
        query = query + "AND EventID > " + latestTimestamp + " order by
    EventID asc ";
    } else {
        // in case it is the first collection cycle, take maxEventsCount
        events from the end
        query = query + " order by EventID desc ";
    }
    return query;
},

ggetEventTimestampFieldName : function () { //return the name of event
timestamp field
    return "swEventId";
},

ggetURL : function (host, query) {
    var port = this.probe.getAdditionalParameter("port"); //retrieve
    all additional parameters unique to this Source
    var url = 'https://'+host+':' + port + '/SolarWinds/
    InformationService/v3/Json/Query?query=' +encodeURIComponent(query);
    return url;
},

createSNEvent : function (rawEvent, eventTypes, applications,
    components, nodes, volumes, interfaces, nodeProperties) { //get all cached
    information as well
    var event = Event();
    event.setSource(SOLAR_WINDS);
    var emsName = this.probe.getParameter("connector_name");
    event.setEmsSystem(emsName); //set the connector instance name as
    source instance
    if (rawEvent.EventTime != null)
        event.setTimeOfEvent(this.parseTimeOfEvent(rawEvent.EventTime));
    // remove not ascii chars
    var sanitizedMessage = rawEvent.Message.replace(/\[\x00-\x7F]/g, "
    ");
    // replace \" with "
    sanitizedMessage = sanitizedMessage.replace(/\"/g, "\n");
if (nodes[rawEvent.NetworkNode] != null) {
    var node = nodes[rawEvent.NetworkNode];
    event.setHostAddress(node[0]);
    event.setField("hostname", node[1]);
}

event.setField("eventType", rawEvent.EventType);

if (eventTypes[rawEvent.EventType] != null) {
    var icon = eventTypes[rawEvent.EventType][0];
    event.setField("icon", icon);
    if (icon === "Add" || icon === "Green" || icon === "Start")
        event.setResolutionState("Closing");
    else
        event.setResolutionState("New");
} else {
    event.setResolutionState("New");
}

this.populateCustomProperties(event, rawEvent, nodeProperties);

    event.setField("netObjectId", rawEvent.NetObjectID);
    event.setField("swEventId", rawEvent.EventID);
    event.setField("networkNodeId", rawEvent.NetworkNode);
    event.setField("NetObjectType", rawEvent.NetObjectType);

    this.populateEventFieldsByNetObjectType(event, rawEvent, components, applications, interfaces, volumes, eventTypes);

    return event;
},

populateEventFieldsByNetObjectType: function (event, rawEvent, components, applications, interfaces, volumes, eventTypes) {

    var NetObjectType = rawEvent.NetObjectID;
    if (NetObjectType.trim() == 'AM') {
        // Network object type is Component - populate Metric Name and Resource fields
        if (components != null) {
            var component = components[rawEvent.NetObjectID];
            if (component != null) {
                event.setMetricName(component[0]);
                if (component[1] != null && applications != null) {
                    var applicationName = applications[component[1]];
                    if (applicationName != null)
                        event.setResource(applicationName);
                }
            }
        }
    } else if (NetObjectType.trim() == 'AA') {
        // Network object type is Application - populate Resource field
        if (applications != null) {
            var application = applications[rawEvent.NetObjectID];
            if (application != null)
                event.setResource(application);
        }
    } else if (NetObjectType.trim() == 'I' || NetObjectType.trim() == 'IW') {
        // Network object type is Interface - populate Metric Name and Resource fields
        if (interfaces != null) {

var interfaceName = interfaces[rawEvent.NetObjectID];
if (interfaceName != null) {
    event.setResource(interfaceName);
    event.setMetricName(eventTypes[rawEvent.EventType][1]);
}
} else if (NetObjectType.trim() == 'V') {
    // Network object type is Volume - populate Metric Name and Resource fields
    if (volumes != null) {
        var volumeName = volumes[rawEvent.NetObjectID];
        if (volumeName != null) {
            event.setResource(volumeName);
            event.setMetricName(eventTypes[rawEvent.EventType][1]);
        }
    }
} // Otherwise, Metric Name and Resource fields will be empty
}
},
populateCustomProperties: function (event, rawEvent, nodesProperties) {
        var nodePropertiesValue = nodesProperties[rawEvent.NetworkNode];
        var propertiesParam = this.probe.getAdditionalParameter("nodes_custom_properties");
        var propertiesArr = propertiesParam.split(',');
        for (var i = 0; i < propertiesArr.length; i++) {
            var currentProperty = propertiesArr[i].trim();
            if (nodePropertiesValue[currentProperty] != null)
                event.setField(currentProperty, nodePropertiesValue[currentProperty]);
        }
    },
},
getResponse: function (query) {
    //return parsed response according to the query type (such as REST or DB);
    return this.getResponseJSON(query, true);
},
getResponseJSON: function(query, reportError) {
    var request = this.createRequest(query);
    request.setHeader('Accept', 'application/json');
    var response = request.get();
    if (reportError && response == null)
        this.addError(request.getErrorMessage());
    return response;
},
getResponseNoError: function(query) {
    return this.getResponseJSON(query, false);
},
createRequest: function(query) {
    var username = this.probe.getParameter("username");
    var password = this.probe.getParameter("password");
    var host = this.probe.getParameter("host");

    var url = this.getUrl(host, query);
    var request = new HTTPRequest(url);
    request.setBasicAuth(username, password);
    return request;
}
updateLastSignature: function(events, retVal) {
    var timeOfEvent = this.getEventTimestampFieldName();
    // the result is sorted, but the sort order can differ. Therefore
    // the last signature is either on the first or the last event
    var firstEventSignature = events[0].getField(timeOfEvent);
    var lastEventSignature =
        events[events.length-1].getField(timeOfEvent);

    if (parseInt(firstEventSignature) >= parseInt(lastEventSignature))
        retVal['last_event'] = firstEventSignature;
    else
        retVal['last_event'] = lastEventSignature;
},
getResult : function (query) {
    var response = this.getResponse(query);

    if (response == null) {
        this.addError("SolarWindJS Connector: Failed to bring data. Null
response");
        return null;
    }

    if (response.getStatusCode() != 200) {
        this.addError("SolarWindJS Connector Error Code: " +
response.getStatusCode());
        return null;
    }

    return this.parseToJSON(response);
},
getResultNoError : function (query) {
    var response = this.getResponseNoError(query);

    if (response == null || response.getStatusCode() != 200) {
        return null;
    }

    return this.parseToJSON(response);
},

//get response and parse it to JSON
parseToJSON : function (response) {
    var parser = new JSONParser();
    var resultJson =  parser.parse(response.getBody() );
    ms.log("SolarWindJS Connector: Found " + resultJson.results.length +
" records");
    return resultJson;
},

parseTimeOfEvent: function (sourceTime) {
    // input is yyyy-MM-dd'T'HH:mm:ss.mmm. we are taking yyyy-MM-dd HH:mm:ss
    var timeOfEvent = sourceTime.replace('T',' ');
    timeOfEvent = timeOfEvent.substring(0,19);
    return timeOfEvent;
getEventTypes : function () {
    var resultJson = this.getResult(
        "SELECT EventType, Name, Icon, NotifyMessage, NotifySubject FROM Orion.EventTypes");
    if (resultJson == null)
        return null;
    var resultMap = {};
    var i = 0;
    for (; i<resultJson.results.length; i++)
        resultMap[resultJson.results[i].EventType] = [resultJson.results[i].Icon, resultJson.results[i].Name];
    return resultMap;
},

getApplications : function () {
    var resultJson = this.getResultNoError("SELECT ApplicationId, Name FROM Orion.Apm.Application");
    if (resultJson == null)
        return null;
    var resultMap = {};
    var i = 0;
    for (; i<resultJson.results.length; i++)
        resultMap[resultJson.results[i].ApplicationId] = resultJson.results[i].Name;
    return resultMap;
},

getComponents : function () {
    var resultJson = this.getResultNoError("SELECT ComponentID, Name, applicationID FROM Orion.Apm.Component");
    if (resultJson == null)
        return null;
    var resultMap = {};
    var i = 0;
    for (; i< resultJson.results.length; i++)
        resultMap[resultJson.results[i].ComponentID] = [resultJson.results[i].Name, resultJson.results[i].applicationID];
    return resultMap;
},

getInterfaces : function () {
    var resultJson = this.getResultNoError("SELECT InterfaceId, InterfaceName FROM Orion.NPM.Interfaces");
    if (resultJson == null)
return null;
var resultMap = {};
var i = 0;
for (; i< resultJson.results.length; i++)
    resultMap[resultJson.results[i].InterfaceId] =
    resultJson.results[i].InterfaceName;
return resultMap;
",
getVolumes : function () {
    var resultJson =
        this.getResultNoError("SELECT VolumeID, Caption FROM Orion.Volumes");
    if (resultJson == null)
        return null;
    var resultMap = {};
    var i = 0;
    for (; i< resultJson.results.length; i++)
        resultMap[resultJson.results[i].VolumeID] =
        [resultJson.results[i].IPAddress,
         resultJson.results[i].DNS];
    return resultMap;
},
getNodes : function () {
    var resultJson = this.getResult(
        "SELECT NodeID, IPAddress, DNS FROM Orion.Nodes");
    if (resultJson == null)
        return null;
    var resultMap = {};
    var i = 0;
    for (; i<resultJson.results.length; i++)
        resultMap[resultJson.results[i].NodeID] =
        [resultJson.results[i].IPAddress,
         resultJson.results[i].DNS];
    return resultMap;
},
getNodesProperties : function () {
    var properties =
        this.probe.getAdditionalParameter("nodes_custom_properties");
    var resultMap = {};
    if (properties) {
        if (properties.startsWith("E.g:")) //no custom value -> no custom
            properties
                return null;
        query = "SELECT NodeID, " + properties + " FROM Orion.NodesCustomProperties";
        var resultJson = this.getResultNoError(query);
        if (resultJson == null)
            return null;
Create a custom pull bi-directional connector

Create a bi-directional connector definition to send changes made to an alert back to the external event source.

Role required: evt_mgmt_admin

Currently, the SCOM connector is the only connector with bi-directional functionality by default. However, you can customize any connector to have bi-directional functionality.

When a connector definition is created, the boilerplate code that it contains is for the exchange of values from the external event source to the ServiceNow instance. Implement bi-directional functionality as one of these options:

- Connector with full abilities, where there is an exchange of values to-and-from the external event source and the ServiceNow instance. Do this implementation by adding the `updateSource` method to the connector code.
- A pull connector with bi-directional functionality. In a case where events are pushed to the ServiceNow instance from the event source (such as using REST or SNMP), the connector must contain an implementation only for forwarding updates back to the event source. Do this implementation by replacing the connector boilerplate code with the code sample in this document.

**Note:** Users must have write permission to the external source to provide the bi-directional functionality.

The MID Server can update any connector when manual changes to alerts are received, as depicted in this diagram.
1. Navigate to **Event Management > Event Connectors (Pull) > Connector Definitions** and click **New**.

2. In the **Connector Definition** form, select the **Bi-directional** option. When the connector instance is configured as bi-directional, each manual update of an alert from this source instance is logged in the Update Queue table (em_connector_update_queue).

3. In the **Alert field identifier** field, specify the alert field that contains the identifier used to associate the ServiceNow alert with the corresponding event or alert in the external event source. This identifier must be an actual alert field, not a field in the **Additional information** of the alert. To include a field that is contained only in the **Additional information** field, see **Custom alert fields**.

   - The **Update queue** business rule on the Alert table identifies each manual update of the alert and updates the connector queue with these changes. By default, changes to all alert fields are tracked.
   - The **Event Management - Queue connector** processor schedule job dequeues alert changes and sends them to the MID Server. By default, this dequeue process is performed in batches of 1,000 alerts. You can configure this batch size using the `evt_mgmt.max_update_source_records` property.

4. In the connector JavaScript, specify the information required for the **updateSource** function. This function receives the manual changes from alerts and sends them to the external source, according to the source instance. Use this **updateSource** function to handle and parse all the alert fields that should be updated in the external source.

5. After adding the update source method, update the test connection method.
Note: When adding bi-directional functionality for a Push connector, the Connector Instance name must match the **Source Instance** field of the alerts generated by that event source.

For an example of a connector script with full connector abilities, see the SCOM connector script that is provided with the base instance.

To match alerts to the specific connector instance in the following example, the manual update of the alert is sent to the correct connector instance by matching the source instance name.

When using the `updateSource` method, specify which fields you are looking to find and send to the source. In this code, you can find an example of **Incident** and **State** fields. Whenever a new incident is associated to the alert or the state of the alert is being changed manually, an update is sent to the external source monitor. External server users must have write permission to the external source to provide the bi-directional functionality. The following script example is of a new custom connector with bi-directional ability only, without retrieving events. The `execute` method must still appear in the script, but it is an implementation that returns success without performing any action.

```javascript
var ConnectorJS = Class.create();

var SUCCESS = Packages.com.service_now.mid.probe.tpcon.OperationStatusType.SUCCESS;
var FAILURE = Packages.com.service_now.mid.probe.tpcon.OperationStatusType.FAILURE;
var Event = Packages.com.snc.commons.eventmgmt.Event;
var SNEventSenderProvider = Packages.com.service_now.mid.probe.event.SNEventSenderProvider;
var HTTPRequest = Packages.com.glide.communications.HTTPRequest;

var errorMessage = "";

ConnectorJS.prototype = Object.extendObject(AProbe, {

    // test the connection with the target monitor
    testConnection : function() {
        ms.log("Connector testing connection");
        var retVal = {};
        try {
            //TODO: run test query
            if (true){ //TODO: validate the request response
                retVal['status'] = SUCCESS.toString();
            } else{
                this.addError(response.getErrorMessage());
                retVal['status'] = FAILURE.toString();
            }

            } catch (e) {
                this.addError(e.toString());
                retVal['status'] = FAILURE.toString();
            }

            ms.log("Connector Connector testConnection " + retVal['status'] )
            if (retVal['status'] == FAILURE.toString())
```

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
```javascript
    retVal['error_message'] = errorMessage;
    return retVal;
},

execute: function() {
    var retVal = {};
    retVal['status'] = SUCCESS.toString();
    return retVal;
},

updateSource : function() {
    ms.log("Bi-directional: Updating the source");
    var jsonObj = this.probe.getAdditionalParameter("alerts"); //get all the alerts data from the Update Queue table
    ms.log("JSON: " + jsonObj);
    jsonObj = jsonObj + '';
    var objJSON = JSON.parse(jsonObj);
    var command = '';
    var argument = '';
    var jsonRes = '';
    var url = this.probe.getAdditionalParameter("url");
    var username = this.probe.getParameter("username");
    var password = this.probe.getParameter("password");
    var host = this.probe.getParameter("host");
    var result = false;
    var retVal = {};
    retVal = {};
    if (objJSON == null){
        this.addError("Received json is empty");
        retVal['status'] = "" + FAILURE.toString();
        retVal['error_message'] = errorMessage;
        return retVal;
    }
    if (url == undefined || url == null || url == '' ) {
        this.addError("URL parameter is empty");
        retVal['status'] = "" + FAILURE.toString();
        retVal['error_message'] = errorMessage;
        return retVal;
    }
    var alertsToSend = [];
    var alertId = 0;
    for (var i = 0; i < objJSON.length; ++i) {
        var alert = objJSON[i];
        for (var j = 0; j < alert.value.length; ++j) {
            if (alert.value[j].fieldName == 'incident' || alert.value[j].fieldName == 'remote_task_id'){
                command = 'ticket_id';
                argument = alert.value[j].newValue;
            } else if (alert.value[j].fieldName == 'state' && alert.value[j].oldValue == 'Open' || alert.value[j].oldValue == 'Reopen' && alert.value[j].newValue == 'Closed'){
                command = 'close';
            } else if (alert.value[j].fieldName == 'state' && alert.value[j].oldValue == 'Closed' && alert.value[j].newValue == 'Open' || alert.value[j].newValue == 'Reopen'){
                ;
            }
```
command = 'open';
}
if (command != ''){
    this.probe.setParameter("action_performed", "true");

    var alertToSend = {};
    alertToSend.id = alert.key;
    alertToSend.command = command;
    alertToSend.ticket_id = argument;

    alertsToSend[alertId] = alertToSend;

    command = ''; 
    argument = ''; 
    jsonRes = JSON.stringify(alertsToSend);
    ms.log("jsonRes: " + jsonRes);
    result = this.sendAlert(url, username, password, jsonRes);
    if (result == false){
        retVal['status'] = "" + FAILURE.toString();
        retVal['error_message'] = errorMessage;
        return retVal;
    }
    alertsToSend = [];
    alertId = 0;
}

retVal['status'] = "" + SUCCESS.toString();
return retVal;
}

sendAlert : function(url, user, pw, body) {
    try {
        var PostMethod = Packages.org.apache.commons.httpclient.methods.PostMethod;
        var GlideHTTPCient = Packages.com.glide.communications.HTTPClient;
        var httpClient = new GlideHTTPCient();

        var postMethod = new PostMethod(url); // put your endpoint URL here
        postMethod.addRequestHeader('X-Auth-Token', '...'); // add request
        headers, can use user and pw parameters for authentication
        postMethod.setRequestBody(body);

        var httpStatus = httpClient.executeMethod(postMethod);
        var response = postMethod.getResponseAsString();
        postMethod.releaseConnection();

        if(response.haveError()) {
            this.addError("ERROR: " + response.getErrorMessage() + " " + 
            response.getBody());
            return false;
        }
        return true;
    }
    catch(err) {
        this.addError("Error on sendAlert function. Failed to send data to the
        source. Error: " + err.message);
        return false;
    }
},

addError : function(message){
    if (errorMessage === "")
        errorMessage = message;
    else
        errorMessage += "\n" + message;
    ms.log(message);
},

Troubleshooting

If manual changes from alerts do not update the external source monitor, perform these checks:

- Check to see if the bi-directional status on the connector is in error.
- Check if changes are written to the queue (em_connector_update_queue) and check for their status.
- Check if changes are being processed on the queue:
  - Yes: Check the MID log for errors.
  - No: Check the ECC queue for processing issues.

Assume that a custom bi-directional connector must be created. The connector must receive event information from an OMi device. When a manual update is made to an alert, the connector must forward the changes back to the OMi external device. Required:

- Write the MID Server custom bi-directional connector script include to enable the exchange of values to-and-from the OMi device external event source and the ServiceNow instance.
  - In the connector JavaScript, specify the information required for the updateSource function. This function receives the manual changes from alerts and sends them to the external source, according to the source instance. Use this updateSource function to handle and parse all the alert fields that should be updated in the external source.
  - After adding the update source method, update the test Connection method to test the bi-directional API as well.
  - Modify an existing connector definition, or create a new connector definition, to which the bi-directional connector script is associated. Thereafter, a custom connector instance is created that uses this connector definition.

In this step, create the custom bi-directional connector definition.


2. Fill in the fields as appropriate, including selecting the Bi-directional option.
   When the connector instance is configured as bi-directional, each manual update of an alert from this source instance is logged in the Update Queue (em_connector_update_queue) table.

3. In the Alert field identifier field, specify the alert field that, if changed manually, causes the external monitoring system to be updated.
The **Update queue** business rule that runs on the Alert table identifies each manual update of the alert and updates the connector queue with these changes. To determine if an update to the alert was caused manually or automatically from an event, the **Last Update Time By Event** field is examined. If this field has not changed, the business rule accepts the alert update as a manual update.

4. Right-click the form header and select **Save**.
5. In the **Script type** field, select **JavaScript**.
6. In the **JavaScript to run** field, select the custom bi-directional connector script include.
7. Click **Update**.

In this step, create the custom bi-directional connector instance and associate the new custom connector definition with it.

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances** and click **New**.
2. Fill in the fields as appropriate. In the **Connector definition** field, select the custom connector definition that you created in the preceding procedure.
3. Select the **Bi-directional** option
4. Right-click the form header and select **Save**.
5. Click **Test Connector**. If a failure result is returned, follow the instructions that are issued by the error to correct the problem and then run another test.
6. When **Test Connector** returns a successful result, click the **Active** option
7. Click **Update**.

Every 30 seconds the **Event Management - Queue connector processor** job runs and processes all pending alerts in the Update Queue. These updates are sent to the MID Server using the **ConnectorUtil** script include.

The Event Management - Queue connector processor schedule job dequeues alert changes and sends them to the MID Server. By default, this dequeue process is performed in batches of 1,000 alerts. You can configure this batch size using the **evt_mgmt.max_update_source_records** property.

---

**Collect raw metric information from a custom metric connector instance**

You can customize a pull metric connector to collect raw metric information from an external source.

The process of collecting raw metric information from a custom metric connector instance is:

- Create a custom metric connector script to collect raw metric information.
- Add the custom metric connector script to a new Event Management connector definition in the **JavaScript to run** field. The connector definition uses the script to enable the MID Server to connect to the external server to obtain metric raw data.
- Define a custom metric connector instance. The custom connector definition is specified in this connector instance.
- The **Event Management - Connector execution job** job executes all connectors in “running mode” and sends a message to the MID Server on the ECC queue.
- The MID Server executes the custom metric connector script.
The custom metric connector script collects raw metric information that it adds to be processed on the MID Server.

- The custom metric connector script returns results to the instance.
- The Event Management – Connector business rule on the ecc_queue table updates the custom metric connector instance with the results, such as, status, error, and signature.

The custom metric connector script must:

- Be written in JavaScript.
- In the connector definition, be specified in the JavaScript to run field.
- Include the retrieveKpi() method. The connector framework uses the retrieveKpi() method to collect metrics. In this method, include the logic for reading data from the external source system and transform to the RawMetric object (in ServiceNow format).
- Read the required connector instance parameters defined in the probe object.
- Collect raw metric data from the metric source.
- Prepare a RawMetric record. The RawMetric object can be created by passing these attributes in the constructor:

<table>
<thead>
<tr>
<th>RawMetric record description</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>metricType</td>
<td>Name of the metric.</td>
</tr>
<tr>
<td>metricTypeId</td>
<td>SysId of the metric, if available. Otherwise pass the value null</td>
</tr>
<tr>
<td>resource</td>
<td>Information about the resource for which metric data is being collected.</td>
</tr>
<tr>
<td>node</td>
<td>IP, FQDN, name of the CI, or host. For example, the name of the Linux server where the disks are installed.</td>
</tr>
<tr>
<td>ci</td>
<td>SysId of the CI if available. Otherwise use the provided information as part of ciIdentifier for the binding.</td>
</tr>
<tr>
<td>value</td>
<td>Value of the metric.</td>
</tr>
<tr>
<td>timestamp</td>
<td>Epoch timestamp of the metric in milliseconds.</td>
</tr>
<tr>
<td>ciIdentifier</td>
<td>List of key-value pairs to identify the CI.</td>
</tr>
<tr>
<td>source</td>
<td>Data source that is monitoring the metric type.</td>
</tr>
<tr>
<td>sourceInstance</td>
<td>Source instance.</td>
</tr>
<tr>
<td>monitoredObjType</td>
<td>Monitored Object Type</td>
</tr>
</tbody>
</table>

- Use the handleMetric function to add the raw metric information from the RawMetric record to be processed in the MID Server. To get access to the MetricHandle object, make the following call to the MetricFactory:

```javascript
var metricHandler = MetricFactory.getMetricHandler()
```

Then for each RawMetric object execute:

```javascript
metricHandler.handleMetric(rawMetric)
```

- Write error logs to the MID Server log using ms.log. You can use the debugLog() function in the MetricCollector script as an example to write debug messages.

Note: You can use the MetricCollector MID Server script include that is provided in the base instance as an example for these items.
The custom script should return this data:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Success or Failure</td>
</tr>
<tr>
<td>error_message</td>
<td>Error Message</td>
</tr>
<tr>
<td>last_kpi_signature</td>
<td>last metric time in milliseconds. This instruction is used for the next query to start from this time. Use any signature that you want to save to retrieve the next metric. This value is updated according to the metric collector Last signature field.</td>
</tr>
<tr>
<td>metric_source</td>
<td>Your source</td>
</tr>
<tr>
<td>metric_count</td>
<td>Count of retrieved metrics</td>
</tr>
<tr>
<td>metric_duration</td>
<td>Duration time of the retrieve metrics process in milliseconds</td>
</tr>
</tbody>
</table>

Test connector requirements - Implement the testConnection() function. The function returns either Success or Failure. If Failure, check the writeError() function for a failure message. You can also check the sa_performance_statistics table to see that metrics are being received. For an example of test connector script, see the SolarWindsJS Mid Server script include.

Metric types that the connector collects:

- Use set active: true/false to define the types you want to collect.
- In the metric collector script use HandleTypes.getActiveTypesForSql(“Your Source”): for a comma-separated list of supported types. If the type is not active, the type is not returned. If the metric source is in registration mode, not type is retrieved. In registration mode, all types are collected to be able to add any missing types.
- You can collect all metrics types. The metricHandler.handleMetric() function filters the types that are collected. However, using only active types in the query commands should improve performance.

Create a custom pull metric connector

A custom pull metric connector requires a script, connector definition, and connector instance to retrieve raw metric information from an external source.

Role required: evt_mgmt_admin

Ensure that the Operational Intelligence plugin is activated, see Request Operational Intelligence.

Prepare the prescribed requirements, see Collect raw metric information from a custom metric connector instance.

The custom metric connector makes remote API calls to an external source to get raw metric data. The connector then writes or sends that data to the instance using the handleMetric() function.

1. Create a custom MID Server script include. As an example, you can use the MetricCollector script that is provided with the base instance.

2. Navigate to MID Server > Script Includes and click New.
   a) In the Name field, specify the name.
   b) Select Active.
c) In the **Script** field, enter the custom metric collector script. Use JavaScript to compose this script.

3. Complete the form as appropriate, and then click **Submit**.
4. Configure a dedicated MID Server. For more information, see *Manually configure a MID Server for Operational Intelligence*.
5. Navigate to **Event Management > Event Connectors (Pull) > Connector Definitions** and click **New**.
   a) In the **Name** field, specify the name.
   b) Select the **Collect metrics** option.
   c) Right-click the form header and select **Save**.
   d) In the **JavaScript to run** field, select the newly created custom MID Server script file.
6. In the **Connector Parameters** area, specify the parameters for all additional information needed for the connector, for example, topics that you want to subscribe to.
7. In the **Connector Definition to MID Server Capabilities** area, specify the details of the MID Server. Use a MID Server with **Operational Intelligence** as a supported application, and with the Metrics capability. For more information, see *Configure a MID Server for Operational Intelligence*.
8. Complete the form as required, then click **Submit**.
9. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances** and click **New**.
   
   **Note:** Remember to specify values for parameters that you defined.
   
   a) In the **Name** field, specify the name.
   b) In the **Connector definition** field, select the newly created custom metric connector definition.
   c) Select the **Metrics collection** option.
   d) In the **Metrics collection schedule** field, specify the number of seconds for the metrics collection schedule job to wait before it repeats.
   e) Right-click the form header and select **Save**.
   f) Complete the form as required, then click **Save**.
10. Click **Test Connector** to verify the connection between the MID Server and the custom metric connector.
11. Click **Active**.
12. Click **Update**.

- To confirm or debug the custom metric connector script, use debug printouts to the MID Server log.
- To monitor incoming events using the custom metric connector instance, navigate to **ECC > Queue** and filter on **ConnectorProbe**.

**Custom connector considerations**

Considerations for custom connector configuration settings when connecting to an external source.
<table>
<thead>
<tr>
<th>Activity</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert information collection</td>
<td>Collect the required information about the alerts.</td>
</tr>
<tr>
<td></td>
<td>• Node – Identifier for host (IP/FQDN, mac, name).</td>
</tr>
<tr>
<td></td>
<td>• Resource – For example, disc, CPU.</td>
</tr>
<tr>
<td></td>
<td>• Message key – The default is <code>source&amp;node&amp;type&amp;resource&amp;metricName</code></td>
</tr>
<tr>
<td></td>
<td>• Severity – If one or more fields in the target monitor must be mapped to ServiceNow severities.</td>
</tr>
<tr>
<td></td>
<td>• Resolution State – If the source monitor has closed alerts, the alerts are closed in the ServiceNow instance.</td>
</tr>
<tr>
<td></td>
<td>• Type - Event type.</td>
</tr>
<tr>
<td></td>
<td>• Metric name - The name and description of the metric to which the alert applies.</td>
</tr>
<tr>
<td></td>
<td>• Additional information – Any other information that might be used to bind the alert to the CI or used for alert action rules, metric value, or alert ID in the target monitor.</td>
</tr>
<tr>
<td>Create ServiceNow events</td>
<td>If an event must be created, ensure that you have the required details. For more information, see <a href="#">Event field format for event collection</a>.</td>
</tr>
<tr>
<td></td>
<td>To populate node, source, event class (ems system), resource, message key, severity, state, type, metric name, additional information and description</td>
</tr>
<tr>
<td></td>
<td>• Source – the name of the monitor</td>
</tr>
<tr>
<td></td>
<td>• Event class = ems system = the name of the connector instance = <code>this.probe.getParameter('connector_name')</code></td>
</tr>
<tr>
<td></td>
<td>• Consider where you can cache information to avoid unnecessary API calls.</td>
</tr>
<tr>
<td>Debugging</td>
<td>Save the debugging information to ms.log. This requires being logged in to the MID Server. An example of a ms.log is:</td>
</tr>
<tr>
<td></td>
<td><code>ms.log('Solarwinds testing connection')</code></td>
</tr>
<tr>
<td></td>
<td>To find the log, search in the MID Server for <code>agent\logs\agent0.log.0</code> file. If there is failure, search the log for more information. The log must reveal the cause of the failure, for example, missing DLLs files or incorrect credentials.</td>
</tr>
<tr>
<td>Execution</td>
<td>• Differentiate between the first pull of events with the subsequent events that are pulled. The first pull of events:</td>
</tr>
<tr>
<td></td>
<td>• Takes events that exist.</td>
</tr>
<tr>
<td></td>
<td>• Must be limited either in time or number of events that are pulled.</td>
</tr>
<tr>
<td></td>
<td>• Must only pull open alerts.</td>
</tr>
<tr>
<td></td>
<td>• Use the main API to get all events data since last signature and limit the number of events.</td>
</tr>
<tr>
<td></td>
<td>• Create ServiceNow events by parsing.</td>
</tr>
<tr>
<td></td>
<td>• Send all event using: <code>var sender = SNEventSenderProvider.getEventSender()</code></td>
</tr>
<tr>
<td></td>
<td>• Update the <code>lastSignature</code> timestamp.</td>
</tr>
<tr>
<td></td>
<td>• Fill the resultant record with the status.</td>
</tr>
</tbody>
</table>
Configure event collection that uses the pull operation
Configure connector instances to use connector definitions (script instructions) that enable the MID Server to connect to external servers to obtain event information.

Use one connector instance for each type of event server that sends events via a MID Server. You can configure connector instances to use the default connector definitions for supported event sources. Alternatively, you can create a custom definition that enables the MID Server to run a script for event collection.

Collect events from external servers using a pull operation

The schedule for the pull interval is set in the Schedule field of the connector instance. Users of the external servers must have read permission. However, if the connector instance is also configured to be bi-directional, the external server users must also have write permission. After configuring the connector instance, use the Test connector to verify the connection with the external source. Thereafter, set the connector instance to Active and events start to be received from the external server according to the pull interval schedule.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last signature</td>
<td>Each time that events are pulled, all events are pulled with the time of that the event occurred, which is &gt;= last signature timestamp. In cases where the last event arrives at the last signature time, it is pulled repeatedly. To prevent this, ‘remember’ the last event ID. Thereafter, when events are next pulled, exclude this event ID if it is in the group of pulled events and its time of event remains unchanged.</td>
</tr>
</tbody>
</table>
| Parameters                       | Determine which parameters are available and how to use them.  
- For context parameters, such as, connector name, user name, password, host, and last event signature. For example: this.probe.getParameter("connector_name").  
- For connector instance values. For example: this.probe.getAdditionalParameter("port"). |
| Target monitor information collection | Collect information about the target monitor.  
- APIs in use, for example, DB or web service.  
- How to connect to the external source:  
  - Credentials.  
  - Parameters (for example, url, port, protocol).  
  - Required Jar or dll files needed to run commands on the MID server. |
| Test connection                  | Verify the connection with the monitor:  
- A valid connection returns a SUCCESS status.  
- Use the main API that returns alerts/events data without sending data.  
- Fill the result record with status information according to the result of the connection test: 
  
  ```java
  retVal['status'] = **SUCCESS.toString()** / FAILURE.toString(). Detailed failure information must be saved to a log.
  ``` |
| Time of event                    | The Time of event is the time that the event occurred. Convert the event time to GMT using the yyyy-MM-dd HH:mm:ss format. |
Connector instances

Configure connector instances to use script instructions that enable the MID Server to connect to external servers to obtain event information. Each connector instance is specific to an event source vendor.

Configure push operation event collection
You can configure listeners to push event information to the instance or MID Server.

Event Management receives external events through an event collector or script. If you are using a script to collect events, no configuration is required. All other methods of collecting events require configuration. For more information, see Event collection using script.

After event collection configuration, Event Management can collect events that are sent directly to the instance, or via the MID Server, SNMP trap, or email.

Standard event form
All events must use a standard event form, regardless of how they arrive at the instance. For more information, see Event field format for event collection.

Event collection via MID Server
The MID Server facilitates communication and movement of data between external applications, data sources, services, and the ServiceNow instance.

<table>
<thead>
<tr>
<th>Event collection modes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provided by the base system</td>
<td>Listeners provided with the Event Management application:</td>
</tr>
<tr>
<td></td>
<td>- AWS CloudWatch, see Integrate AWS platform as a data source.</td>
</tr>
<tr>
<td></td>
<td>- Azure event messages, see Integrate Azure platform as a data source.</td>
</tr>
<tr>
<td></td>
<td>- BMC TrueSight events, see Event collection from BMC TrueSight.</td>
</tr>
<tr>
<td></td>
<td>- Generic event collection through the MID Server.</td>
</tr>
<tr>
<td></td>
<td>- Generic event collection through the instance.</td>
</tr>
<tr>
<td></td>
<td>Where event messages are received in XML, you can base your script on the example xml transform script provided with the base system.</td>
</tr>
<tr>
<td>Via MID WebService</td>
<td>You can configure the MID WebService Event Collector to push event messages to the MID Server. For more information, see Event collection to MID Server using web service API.</td>
</tr>
<tr>
<td>Create your own content</td>
<td>You can configure your own listener:</td>
</tr>
<tr>
<td></td>
<td>- Via MID WebServer API. See Event collection to the instance using web service API.</td>
</tr>
<tr>
<td></td>
<td>- Via Script. See Event collection using script.</td>
</tr>
<tr>
<td></td>
<td>- Via Custom Transformation. See Event collection from custom payloads.</td>
</tr>
<tr>
<td>Via SNMP</td>
<td>SNMP listeners provided with the Event Management application:</td>
</tr>
<tr>
<td></td>
<td>- Generic SNMPv2. See Configure event collection for SNMP traps.</td>
</tr>
<tr>
<td></td>
<td>In addition, you can create your own custom listener. See Configure listener transform scripts.</td>
</tr>
<tr>
<td>Via Email</td>
<td>Email listener. See Configure event collection from email.</td>
</tr>
</tbody>
</table>

Configure a MID Server for event collection using a push operation (listener)
The MID Server supports the collection of event messages, using the MID Web Server to collect data from external sources and transforming (parsing) them to the format required for event fields in the event (em_event) table. The transformed events are then transmitted to the instance.

Role required: evt_mgmt_admin

The MID Web Server extension enables external clients to push metric data and events to the MID Server. See Configure the MID Web Server extension. Use the MID WebService Event Collector to enable a URL method to push event messages to the MID Server. See Configure the MID WebService Event Collector Context.

Configure a MID Server with either EventManagement or All in the Supported Applications List.

To ensure uninterrupted services, consider also configuring a failover MID Server cluster.

1. Navigate to MID Server > Servers.
2. Double-click the MID Server that you want to configure, or click New to create a new MID Server.
3. In the Supported Applications section, click Edit.
   a) Click Edit.
   b) In the slushbucket, either select Event Management or ALL and click the > add button.
   c) Click Save.
4. Ensure that the MID Server is validated. Under Related Links, click Validate.
   For more information, see Validate a MID Server.
5. Click Update.

Configure the MID WebService Event Collector to enable a URL method to push event messages to the MID Server. For more information, see Configure the MID WebService Event Collector Context.

Event collection to MID Server using web service API

Configure the MID WebService Event Collector to provide a URL method to push event messages to the MID Server.

The MID WebService Event Collector, by default, enables you to connect to an event generator and send event messages through the MID Server, using a URL in the format: http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/jsonv2

The JSON v2 format is the same as the format that clients use to send event messages to the instance. This URL provides good performance.

For more information, see Configure the MID WebService Event Collector Context.

Note: Event Management accepts events in the timezone of the MID Server through which the listener has sent the event message. If you require a different timezone, you can configure the script as required, see Configure listener transform scripts.

URL to push custom payloads

The MID Server can also receive a custom payload in JSON, XML, or plain text format. In this case, the MID Server transforms the event messages using the script include that you provide, and then sends the events to the instance. The URL in this case is:

http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/inbound_event?
Transform={Transform_script_name} For an example of a script include, see the default TransformEvents_xmlSample script include.
For an example of how the MID WebService Event Collector transforms JSON formatted event messages, see Event collection from BMC TrueSight.

To send events in generic JSON format, you can use the URL in this format:

http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/inbound_event?Transform=TransformEvents_MidJsonGeneric

In this case, the TransformEvents_MidJsonGeneric MID script include that is provided in the base instance transforms the event messages into the event fields. All fields are added as Additional Information fields into the ServiceNow event that is generated.

Note: The URL in the format http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/{transform_script_name} is also supported.

Script include to transform collected events

Use the required MID Server script include to transform (parse) collected event messages and populate them into the mapped event field. The MID Server script include name is composed of a prefix and suffix. The prefix is the mandatory text TransformEvents_ and the suffix is the transform name on the URL, for example, http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/xmlSample. The composed script name is therefore TransformEvents_xmlSample.

You must supply a script include that receives the event messages in a text variable. The script must have a transform() function that performs the transformation and prepares the return array of Event objects. When the Event Management plugin is activated, you are provided with the TransformEvents_xmlSample sample include script that transforms events from XML format.

Note: Event fields that are not identical to fields in the event table are saved in the Additional information field.

Event message headers

Specify the relevant request header value for the Content-Type field according to the format of the event message.

Request header values for the Content-Type

<table>
<thead>
<tr>
<th>Message format</th>
<th>Content-Type value</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSON</td>
<td>application/json</td>
</tr>
<tr>
<td>XML</td>
<td>application/xml</td>
</tr>
<tr>
<td>text</td>
<td>text/plain</td>
</tr>
</tbody>
</table>

Authentication and data security options

Options for authentication and data security:

- **Authentication type**, which can be set to either Basic or the more advanced Keybased option.
- **Secure Connection**, which lets you choose whether incoming and outgoing data is secured when transmitted. If you choose the advanced secured option, it requires that you obtain a
Users must supply a script that receives the JSON, XML, or text formatted event messages in a text variable. The script must have a `transform()` function that performs the transformation and prepares the return array of event objects. The `TransformEvents_xmlSample` include script that transforms events from XML format is provided as an example when the Event Management plugin is activated.

For more information about authentication and data security, see [Configure the MID Web Server extension](#).

**Event collection to the instance using web service API**

You can use a web service interface, supported by ServiceNow, that operates on the JSON object as the data input and output format.

**Role required:** `evt_mgmt_integration`

To insert records in the event (`em_event`) table with a single call, use this web service API:

```
https://<instancename>.service-now.com/api/global/em/jsonv2
```

**Note:** Business rules on the event table are not invoked when this URL is used.

Do not add additional fields to an event by adding a custom field to the event table (`em_event`). However, additional fields should be included in the `Additional information` (`additional_info`) field of the event. For more information about how to include additional fields in events, see [Custom alert fields](#).

The URL in the format: `http://<instancename>.service-now.com/em_event.do?JSONv2&sysparm_action=insertMultiple` is also supported. However, the performance of the first URL is superior.

To work with the URL that ends with `=insertMultiple`, add the following properties, with their respective values, to the MID Server:

- `"mid.probe.event.queue.compress"` - value: `false`
- `"mid.probe.event.bulk_size"` - value: `100`
- `"mid.probe.event.endpoint.url"` - value: `em_event.do?JSONv2%26sysparm_action=insertMultiple`

1. Send the request with these headers:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>String</td>
<td>The acceptable type for this message.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The default value is <code>application/json</code>.</td>
</tr>
<tr>
<td>Content-Type</td>
<td>String</td>
<td>The content type for this message.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The default value is <code>application/json</code>.</td>
</tr>
<tr>
<td>POST</td>
<td>String</td>
<td>The request type is POST, with one or more trailing records.</td>
</tr>
</tbody>
</table>

2. One or more events in JSON format can be sent as the payload of the web service call. Event fields that should be populated are:
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>The name of the event source type. For example, SCOM or SolarWinds.</td>
</tr>
<tr>
<td>Source Instance (event_class)</td>
<td>Specific instance of the source. For example, SCOM 2012 on 10.20.30.40.</td>
</tr>
<tr>
<td>node</td>
<td>The <code>node</code> field should contain an identifier for the Host (Server/switch/Router/etc.) that the event was triggered for. The value of the <code>node</code> field can be one of the following identifiers of the Host:</td>
</tr>
<tr>
<td></td>
<td>· Name</td>
</tr>
<tr>
<td></td>
<td>· FQDN</td>
</tr>
<tr>
<td></td>
<td>· IP</td>
</tr>
<tr>
<td></td>
<td>· Mac Address</td>
</tr>
<tr>
<td></td>
<td>If it exists in the CMDB, this value is also used to bind the event to the corresponding ServiceNow CI.</td>
</tr>
<tr>
<td>resource</td>
<td>If the event refers to a device, such as, Disk, CPU, or Network Adapter, or to an application or service running on a Host, the name of the device or application must be populated in this field. For example, Disk C:\ or Nic 001 or Trade web application.</td>
</tr>
<tr>
<td>metric_name</td>
<td>Name of the metric that triggered the alert. For example, Used Memory or Total CPU utilization.</td>
</tr>
<tr>
<td>type</td>
<td>The type of event. This type might be similar to the <code>metric_name</code> field, but is used for general grouping of event types.</td>
</tr>
<tr>
<td>message_key</td>
<td>This value is used for de-duplication of events. For example, there might be two events for the same CI, where one event has CPU of 50% and the next event has CPU of 99%. Where both events must be mapped to the same ServiceNow alert, they should have the same message key. The field can be left empty, in which case the field value defaults to <code>source+node+type+resource+metric_name</code>. The <code>message_key</code> should be populated only when there is a better identifier than the default.</td>
</tr>
<tr>
<td>severity</td>
<td>Severity of the event. ServiceNow values for severity range from 1 – Critical to 5 – Info, with the severity of 0 – Clear. Original severity values should be sent as part of the additional information.</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>additional_info</td>
<td>This field is in JSON key/value format, and is meant to contain any information that might be of use to the user. It does not map to a pre-defined ServiceNow event field. Examples include IDs of objects in the event source, event priority (if it is not the same as severity), assignment group information, and so on. Values in the Additional information field of an Event that are not in JSON key/value format are normalized to JSON format when the event is processed.</td>
</tr>
<tr>
<td>time_of_event</td>
<td>Time when the event occurred on the event origin. The format is: yyyy-MM-dd HH:mm:ss GMT</td>
</tr>
<tr>
<td>resolution_state</td>
<td>Optional – To indicate that an event has been resolved or no longer occurring, some event monitors use ‘clear’ severity, while other event monitors use a ‘close’ value for severity. This field is used for those monitors proffering the latter. Valid values are New and Closing.</td>
</tr>
</tbody>
</table>

3. To create multiple records with a single call, trigger the event web service using the following URL, where the <instance name> variable is replaced with the name of the required instance:

https://<instancename>.service-now.com/api/global/em/jsonv2

Example showing the payload for two events that are sent in a single web service call:

```json
{
  "records": [
    {
      "source":"SCOM",
      "event_class":"SCOM 2012 on scom.server.com",
      "resource":"D:",
      "node":"name.of.node.com",
      "metric_name":"Percentage Logical Disk Free Space",
      "type":"Disk space",
      "severity":"4",
      "description":"The disk D: on computer V-W2K8-abc.abc.com is running out of disk space. The value that exceeded the threshold is 38% free space.",
      "additional_info":{
        'scom-severity':'Medium',
        'metric-value':'38',
        'os_type':'Windows.Server.2008'
      }
    },
    {
      "source":"SCOM",
      "event_class":"SCOM 2012 on scom.server.com",
      "resource":"MSSQL-database-name",
      "node":"other.node.com",
      "metric_name":"DB Allocated Size (MB)",
      "type":"Database Storage",
      "severity":"3",
      "description":"High number of active connections for MSSQL-database-name running on name.of.node.com. Active connections exceed 5000.",
      "additional_info":{
        'scom-severity':'High',
      }
    }
  ]
}
```
4. To create one record with a single call, trigger the event web service using the following URL, where the `<instancename>` variable is replaced with the name of the required instance:

http://<instancename>.service-now.com/api/global/em/jsonv2

**Note:** Use of this URL is limited as far as the rate of events it can support.

Example showing the payload for one event that is sent in a single web service call:

```json
{
  "records": [
    {
      "source": "SCOM",
      "event_class": "SCOM 2007 on scom.server.com",
      "resource": "C:",
      "node": "name.of.node.com",
      "metric_name": "Percentage Logical Disk Free Space",
      "type": "Disk space",
      "severity": "4",
      "description": "The disk C: on computer V-W2K8-dfg.dfg.com is running out of disk space. The value that exceeded the threshold is 41% free space.",
      "additional_info": {
        "scom-severity": "Medium",
        "metric-value": "41",
        "os_type": "Windows.Server.2008"
      }
    }
  ]
}
```

Configure the MID Web Server extension

The MID Web Server is an extension that enables external clients to push metric data and events to the MID Server. This extension is used to listen for raw metric data and external events and it provides options for authentication and data security. The raw data is collected by the MID Server and then transmitted to the instance.

If you plan to use Operational Intelligence data, then see Get started with Operational Intelligence for the high level steps of setting up Operational Intelligence.

Role required: evt_mgmt_admin

- Deploy and start a MID Server.
- Enable the MID Web Server (com.snc.sa.mid.webserver) plugin.
- Select the Enable REST Listener option in the Operational Intelligence Metrics Context. For more information, see Manually configure the Operational Intelligence extension.
- If the Secure Connection option is going to be selected, first obtain a server certificate. For more information, see Setup certificate for secure connection.

The MID Web Server extension runs for as long as it is enabled. The extension starts a web server on the MID Server to serve web requests from external systems. The raw data is pushed to the extension from a client or using customized script.
The configuration of the MID Web Server extension includes these settings:

- **Authentication type**, which can be set to the more advanced option - **Keybased**.
- **Secure Connection**, which lets you choose whether incoming and outgoing data is secured when transmitted. If you choose the advanced secured option, it requires that you obtain a certificate from a well-known certificate authority, and then provide the **Keystore Certificate Alias** and the **Keystore Password**.

1. Navigate to **MID Server > Extensions > MID Web Server**.
2. In the MID Web Server Contexts list, click **New**.
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for this MID Web Server collector for easy identification.</td>
</tr>
<tr>
<td>Short description</td>
<td>Enter a brief, meaningful description of this collector.</td>
</tr>
<tr>
<td>Status</td>
<td>This field is auto-populated with the status of the extension. The field is blank until the extension is started. After issuing a command to the extension, one of the following values is displayed:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Started</strong>: The extension is running.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Stopped</strong>: The extension is not running.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Offline</strong>: The MID Server is down.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error</strong>: The extension failed with an error (the error message is displayed in <strong>Error Message</strong>).</td>
</tr>
<tr>
<td></td>
<td>- <strong>Warning</strong>: A run-time exception has occurred. The extension continues to work.</td>
</tr>
<tr>
<td>HTTP/HTTPS Port</td>
<td>Port number on which you want to listen to incoming requests.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Authentication Type</td>
<td>Select one:&lt;br&gt;&lt;br&gt;&lt;strong&gt;Keybased&lt;/strong&gt;&lt;br&gt;- Create an authentication token that is sent with each request.&lt;br&gt;- Send this authentication token in the request header &lt;strong&gt;Authorization&lt;/strong&gt;.&lt;br&gt;&lt;br&gt;To create an authentication token:&lt;br&gt;1. Construct a string using defined elements of the HTTP/HTTPS request.&lt;br&gt;2. Create a Hash Message Authentication Code (HMAC) of the string, that is, sign the string generated in previous step with the auto-generated secret key. The key is unique per context.&lt;br&gt;&lt;br&gt;See example, <a href="#">Manually configure MID Web Server for metric data</a>.&lt;br&gt;&lt;br&gt;&lt;strong&gt;Note&lt;/strong&gt;: A valid timestamp (using the HTTP Date header) is required for the authenticated request. In addition, the timestamp must be within 15 minutes of the time on the MID Server.</td>
</tr>
<tr>
<td>Basic</td>
<td>- The user must provide a username and password. The same username and password must be provided for every request.&lt;br&gt;- On the instance, the password is stored encrypted and it is sent also to the MID Server encrypted.&lt;br&gt;- In the MID Server, the password is saved in memory.&lt;br&gt;- When the request is received, the password is decrypted and matched with the password provided in the request.</td>
</tr>
<tr>
<td>Secret Key</td>
<td>The value that is generated when keybased authentication is selected for the Authentication Type field.</td>
</tr>
<tr>
<td>Error Message</td>
<td>Message describing any error that causes a command, such as Start or Stop, to fail. This field only appears when the value in the Status field is Error.</td>
</tr>
<tr>
<td>Execute on</td>
<td>Location for running this extension. The available options are Specific MID Server or Specific MID Server Cluster.</td>
</tr>
</tbody>
</table>
### Field Description

**MID Server**

Depending on your selection in **Execute on**, the name of the designated MID Server, or MID Server cluster respectively:

- If you selected **Specific MID Server**, the name of the designated MID Server.
- If you selected **Specific MID Server Cluster**, the name of the designated MID Server cluster.

The MID Web Server extension supports MID Server clusters that are configured for failover. If you selected the MID Server cluster option, an algorithm determines which MID Server in the cluster runs the extension. The extension can run on only one MID Server. If the MID Server in the cluster that runs the MID Web Server extension goes down, the extension automatically starts to run on the secondary MID Server. The secondary is the MID Server that is up when the primary goes down.

**Note:** Ensure that in the Supported Applications section of the MID Server **ALL applications** is selected.

**Executing on**

The name of the MID Server on which the extension is running. This field shows the name of the MID Server even if the MID Server is down. If the user stops the extension, this field is empty.

### Commands in the MID Web Server form

<table>
<thead>
<tr>
<th>Related Link</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>If it is not running, starts the MID Web Server.</td>
</tr>
<tr>
<td>Stop</td>
<td>Stops the MID Web Server. If the collector is not running, no action is taken.</td>
</tr>
<tr>
<td>Restart</td>
<td>Stops, then starts the MID Web Server.</td>
</tr>
<tr>
<td>Test parameters</td>
<td>The test is not relevant to the MID Web Server. Parameters are not tested or validated when <strong>Test</strong> is run.</td>
</tr>
</tbody>
</table>
For an example that describes how to prepare an authentication token, see [Manually configure MID Web Server for metric data](#).

Configure the MID WebService Event Collector Context

Configure the MID WebService Event Collector Context to provide a URL method to push event messages from an external source to the MID Server.

Role required: evt_mgmt_admin

The default format of the URL to push event messages from an external source to the MID Server is http://{MID_Server_IP}:{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/jsonv2. This URL provides good performance.

From an external source, to push event messages that are not in jsonv2 format, the format of the URL is: http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/inbound_event?Transform={Name_of_Transform_Script}, where the {Name_of_Transform_Script} variable is the full name of the script and always begins with the text: TransformEvents_.

For example, assume the following values:

- {MID_Server_IP}: 10.118.69.27
- {MID_Web_Server_Port}: 8097
- Transform script name: EventsToProcess

The URL to be used is therefore: http://10.118.69.27:8097/api/mid/em/inbound_event/TransformEvents_EventsToProcess

**Note:** The URL in the format http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/{transform_script_name} is also supported.

**Note:** The date format for events is yyyy-M-d h:mm:ss.

If you receive an event whose date is in a different format, you must use a {transform_script_name} that is appropriate for the incoming event's date format. If you do not, the event will not be processed correctly.

For example, if an event arrives on June 27, 2019 at 11:25 AM with a listed date of 2019/06/27/11:25:00 a.m., use a {transform_script_name} with a date format of yyyy/MM/dd/ HH:mm:ss a to match the format of the received event.

2. In the MID WebService Event Collector Contexts list, click New.
3. On the form, fill in the fields.

### MID WebService Event Collector Context form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for this collector for easy identification.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Short description | Enter a brief, meaningful description of this collector.

MID Web Server Extension | Specify and then start the MID Web Server extension. The supported authentication methods are listed in the **Authentication Type** field of the MID Web Server extension. For information about how to configure a MID Web Server extension, see [Configure the MID Web Server](#).

Status | This field is auto-populated with the status of the MID Web Server extension. This field is blank until the MID Web Server extension is started. After issuing a command to the MID Web Server extension, one of the following values is displayed:
- **Started**: The collector is running.
- **Stopped**: The collector is not running.
- **Offline**: The MID Server is down.
- **Error**: The collector failed with an error (the error message is displayed in **Error Message**).
- **Warning**: A run-time exception has occurred.

Execute on | **Specific MID Server** or **Specific MID Server Cluster**, as defined on the specified MID Web Server extension.

MID Server | The **Specific MID Server** or **Specific MID Server Cluster**, as defined on the specified MID Web Server extension.

Executing on | The name of the MID Server on which the MID Web Server extension is running.

---

4. Right-click the form heading and click **Save**.
5. Under **Related Links**, click **Start** to start the collector.

### Commands in the Event Management Context form

<table>
<thead>
<tr>
<th>Related Link</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>If it is not running, start the collector. This action verifies that a web service API endpoint with the Event Management application is running on the MID Server.</td>
</tr>
<tr>
<td>Stop</td>
<td>Stops the running collector on the configured MID Server. If the collector is not running, no action is taken.</td>
</tr>
<tr>
<td>Restart</td>
<td>Stops, then starts the collector on the configured MID Server.</td>
</tr>
</tbody>
</table>
Showing the use of the URL to transform JSON v2 formatted event messages

Assume that JSON v2 formatted event messages are sent to the MID Server. When using the `jsonv2` URL, there is no need to use a script include.

**Data for the JSON v2 example**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MID_Server_IP</td>
<td>10.218.64.27</td>
</tr>
<tr>
<td>MID_Web_Server_Extension_Port</td>
<td>8097</td>
</tr>
<tr>
<td>Event message format</td>
<td>jsonv2</td>
</tr>
</tbody>
</table>

Replace the variables in the default format of the URL `http://<my-instance>.service-now.com/api/global/em/jsonv2` with values from the preceding table:`http://10.218.64.27:8097/api/global/em/jsonv2`

**Example showing the URL to push messages not in jsonv2 format**

The format of the URL to push event messages from an external source that are not in jsonv2 format is `http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/inbound_event/Transform={Name_of_Transform_Script}` where the `{Name_of_Transform_Script}` variable is the full name of the script and always begins with the text: TransformEvents_. The script name must be specified as the Transform header parameter and must always start with the prefix TransformEvents_.

For this example, assume that the script name is EventsToProcess, the URL is therefore:`http://10.138.64.27:8097/api/mid/em/inbound_event/TransformEvents_EventsToProcess`

**Example showing JSON v2 formatted event messages**

**Note:** When copying and pasting the text following, hidden characters might also be copied and can cause unexpected results.

```
```
Event collection from BMC TrueSight

The MID WebService Event Collector enables you to collect JSON formatted event messages sent from BMC TrueSight Operations Management (TrueSight), previously known as BMC ProactiveNet Performance Management (BPPM), utilizing event stream notification capabilities.

Ensure that you:

- Deploy and start the MID Server. See Configuring MID Servers.
- Configure and start the MID Web Server. See Configure the MID Web Server extension.
- Configure and start the MID WebService Event Collector. See Configure the MID WebService Event Collector Context.

Role required: evt_mgmt_admin

JSON formatted event messages are sent from BMC TrueSight. The MID Server transforms the collected event messages by parsing them using the TransformEvents_bmcTrueSight script include, located here: Event Management > Event Listener (Push) > Listener Transform Scripts. In the Listener Transform Scripts page, click TransformEvents_bmcTrueSight.

The default format of the URL to push event messages from BMC TrueSight to the MID Server is http://<MID_Server_IP>:<MID_Web_Server_Port>/api/mid/em/inbound_event?Transform=TransformEvents_bmcTrueSight.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MID_Server_IP</td>
<td>IP address of the MID Web Server Extension.</td>
</tr>
<tr>
<td>MID_Web_Server_Port</td>
<td>Listening port of the MID Web Server Extension.</td>
</tr>
<tr>
<td>MID_Web_Server_User</td>
<td>Username for the MID Web Server Extension.</td>
</tr>
<tr>
<td>MID_Web_Server_Password</td>
<td>Password of the user of the MID Web Server Extension. However, encode the password in BASE64. For example, if the MID Web Server Extension password is admin, in this case use YWRtaW4= as the password for MID_Web_Server_Password. To determine the BASE64 value, you can use the utility at this website: <a href="https://www.base64encode.org/">https://www.base64encode.org/</a></td>
</tr>
</tbody>
</table>
The following procedure describes the collection of JSON formatted event messages using basic authentication. For more information about supported authentication methods, see Configure the MID Web Server extension.

1. In BMC TrueSight, set the MID WebService Event Collector as the target for BMC TrueSight notification, using these parameters:

   **Variables in the generic BMC URL**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>serverHost</td>
<td>Name of the BMC TrueSight server.</td>
</tr>
<tr>
<td>port</td>
<td>Number of the BMC TrueSight port.</td>
</tr>
</tbody>
</table>

2. In the message body and generic BMC URL, replace the parameters with the values in the preceding table and connect to BMC TrueSight. Use the BMC Publish-Subscribe REST approach to create a stream event subscription and send event messages in JSON format. The generic format of the BMC URL to create a stream event subscription is:


   For example: http://TrueSight1server:40/bppmws/api/stream/subscribe?dataType=event.

**Example of mapping event messages to event fields in the Event (em_event) table**

<table>
<thead>
<tr>
<th>Event field</th>
<th>Event message</th>
</tr>
</thead>
<tbody>
<tr>
<td>node</td>
<td>mc_host</td>
</tr>
<tr>
<td>type</td>
<td>mc_object_class</td>
</tr>
<tr>
<td>metric_name</td>
<td>mc_parameter</td>
</tr>
<tr>
<td>resource</td>
<td>mc_object</td>
</tr>
<tr>
<td>severity</td>
<td>severity</td>
</tr>
<tr>
<td>time_of_event</td>
<td>date_reception</td>
</tr>
<tr>
<td>description</td>
<td>msg</td>
</tr>
</tbody>
</table>

**Example, showing the message body.**

```json
{
  action: 'create',
  data:
  {
    mc_abstracted: [],
    itsm_operational_category3: '',
    mc_object_class: 'CPU',
    source: 'MySource',
    adapter_host: '',
    itsm_operational_category2: '',
    itsm_location: '',
    itsm_operational_category1: '',
    mc_abstraction: [],
    mc_tool_id: '',
    mc_original_priority: 'PRIORITY_1',
    event_handle: '3172',
    mc_modhist: [],
    mc_tool_class: '',
    mc_origin: '',
  }
}
```
Event collection from custom payloads

The MID WebService Event Collector enables you to collect event information from custom payloads in JSON, XML, or plain text format.

Role required: evt_mgmt_admin

The MID Server transforms the collected event messages and populates the Event table fields (em_table) in the instance. The collection of formatted event messages is described in this procedure using basic authentication. For information about supported authentication methods, see Configure the MID Web Server extension.

The format of the required MID Server URL is: http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/inbound_event?Transform={Transform_script_name}. The name of the MID Server script include is composed by appending a suffix to the default TransformEvents_ prefix. For information about the collection of JSON v2 event messages, see Configure the MID WebService Event Collector Context.

Note: The URL in the format http://{MID_Server_IP}:{MID_Web_Server_Port}/api/mid/em/{transform_script_name} is also supported.

1. Configure the MID WebService Event Collector, see Configure the MID WebService Event Collector Context.

2. Start the MID WebService Event Collector.

Transformation of XML formatted event messages using the custom payload URL

Assume that XML formatted event messages are sent to the MID Server. Use this example to return an array of event objects from the collected event messages. The name of the MID Server script include is composed by appending a suffix to the default TransformEvents_ prefix. For the purposes of this example, the user supplied the xmlSample script include. Using these details, the name of the MID Server script include is TransformEvents_xmlSample. The MID Server transforms
the collected event messages by parsing the messages using the script include and then transmitting them to the instance.

### Data for the XML example

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MID_Server_IP</td>
<td>10.218.64.27</td>
</tr>
<tr>
<td>MID_Web_Server_Extension_Port</td>
<td>8097</td>
</tr>
<tr>
<td>transform_script_suffix_name</td>
<td>xmlSample</td>
</tr>
</tbody>
</table>

Replace the variables in the URL with the values from the above table:

```text
http://10.218.64.27:8097/api/mid/em/xmlSample
```

**Note:** When copying and pasting the text below, hidden characters might also be copied and can cause unexpected results.

**Example showing XML formatted event messages:**

```xml
<records>
  <event>
    <source>My Source</source>
    <node>host1</node>
    <type>type1</type>
    <severity>3</severity>
    <description>Virtual memory usage exceeds 98%</description>
  </event>
  <event>
    <source>My Source</source>
    <node>host2</node>
    <type>type2</type>
    <severity>2</severity>
    <description>Virtual memory usage exceeds 90%</description>
  </event>
</records>
```

### Event collection using script

You can create a scripted integration with Event Management to push external events from monitoring tools. You can use the cURL command-line tool to send one or more events to a ServiceNow instance.

For Mac, cURL is automatically available.

For Unix, Linux, or Windows, make sure that the cURL command-line tool is installed. You can download it at [http://curl.haxx.se/download.html](http://curl.haxx.se/download.html).

**Role required:** evt_mgmt_integration

For scripted integration, no configuration is required. If your monitoring tool is a supported event source, you can instead select a connector and configure an instance to pull events using a MID Server. In either case, the system uses a web service API to add events to the Event (em_event) table, see [Event collection to MID Server using web service API](#).

1. Use a text editor to create a shell (.sh) file with event records.

```bash
curl -v -H "Accept: application/json" -H "Content-Type: application/json"
-X POST --data "{
```

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
"records": [
    {
        "source": "Simulated",
        "node": "nameofnode",
        "type": "High Virtual Memory",
        "resource": "C:",
        "severity": "5",
        "description": "Virtual memory usage exceeds 98%",
        "ci_type": "cmdb_ci_app_server_tomcat",
        "additional_info": "{\"name\":\"My Airlines\"}\"
    },
    {
        "source": "Simulated",
        "node": "01.myairlines.com",
        "type": "High CPU Utilization",
        "resource": "D:",
        "severity": "5",
        "description": "CPU on 01.my.com at 60%"
    }
]
" -u myUserID:myPassword https://<my-instance>.service-now.com/api/
global/em/jsonv2

2. Use the -H option. For the POST parameter, start the data block with an open bracket and
   delimit the data with backslashes. For example:

   POST --data "({"records":[{"source": [. . .]}})"

   Header parameters for use with -H option

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>String</td>
<td>The acceptable type for this message. The default value is application/json.</td>
</tr>
<tr>
<td>Content-Type</td>
<td>String</td>
<td>The content type for this message. The default value is application/json.</td>
</tr>
<tr>
<td>POST</td>
<td>String</td>
<td>The request type is POST with one or more trailing records.</td>
</tr>
</tbody>
</table>

3. Use the -u option and the instance URL with login credentials. For example:

   -u myUserID:myPassword "https://<my-instance>.service-now.com/api/
global/em/jsonv2"

4. Test the scripted integration.

Configure listener transform scripts
Configure a push connector to connect to an external event source, using custom script that
processes the collected event messages and transforms it to the required event format. Select
to send events either through the MID Server or the instance, in each case using the URL of the
required format.

Role required: evt_mgmt_admin
The listener transform script accepts event messages that are generated by external event sources.

Configure the connector to listen to an external event source. Using custom listener transform script, send the event messages through either the MID Server or the instance.

**Note:** You can use this generic JSON target URL to collect events: https://<<INSTANCE>>/api/global/em/inbound_event?source=genericJson. This URL can be used as-is and requires an event rule to be configured.

1. Navigate to **Event Management > Event Listener (Push) > Listener Transform Scripts**.
2. Click **New** or click the listener transform script that you want to modify, for example, AWS or Azure.
3. Fill in the fields in the form, as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for this listener for easy identification.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for the use of the listener transform script.</td>
</tr>
<tr>
<td>Header name</td>
<td>Specify the name of the header that you want the value for, such as Set-Header.</td>
</tr>
<tr>
<td>Header value</td>
<td>The value of the header name that you want the value for, such as Set-Header.</td>
</tr>
<tr>
<td>Type</td>
<td>Select to send events either <strong>Instance</strong> or <strong>MID</strong>. In each case, use the URL of the selected type. See the description of the <strong>URL parameter value</strong> field.</td>
</tr>
<tr>
<td>Active</td>
<td>Select to enable pulling events from this external event source.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which an event rule is evaluated when multiple rules are defined for the same type of event. Event rules are evaluated in ascending order.</td>
</tr>
<tr>
<td>URL parameter value</td>
<td>• Send events to the instance.</td>
</tr>
<tr>
<td></td>
<td>Use the URL in this format for Azure: https://&lt;&lt;INSTANCE&gt;&gt;/api/global/em/inbound_event_azure. Use the URL in this format for all connectors, other than Azure: https://&lt;&lt;INSTANCE&gt;&gt;/api/global/em/inbound_event?source={URL_parameter_value}</td>
</tr>
<tr>
<td></td>
<td>• Send events through the MID Server.</td>
</tr>
<tr>
<td></td>
<td>Use the URL in the format: http://(MID_Server_IP):(MID_Web_Server_Port)/api/mid/em/inbound_event?Transform={Transform_script}</td>
</tr>
</tbody>
</table>

**Note:** Sending events to the MID Server requires the prior configuration of the MID WebService Event Listener extension.
4. In the **Script** section:
   - If the value selected for the **Type** field is **MID**, the **Transform script** field appears. In this field, specify or search for the name of the MID script include that accepts event messages that the required external event source generates and that the script parses into the required event format. Use this naming convention for the script: `TransformEvents_<your_source>`
   - If the value selected for the **Type** field is **Instance**, the **Script** editor appears. In the **Script** editor, enter the customized script that accepts event messages that the required external event source generates and that the script parses into the required event format.

Example, showing fields that have been transformed, being added to an event form.

```javascript
(function process(/*RESTAPIRequest*/ request, body) {
/*Function that receives a JSON object, adding all its fields to the Additional information object. The field name is a concatenation of the field key and the parent field key if it exists.*/
function updateAdditionalInfo(event, field,jsonObject,additionalInfo) {
    for (var key in jsonObject) {
        var newKey = key;
        if (field != "") {
            newKey = field + '_' + key;
        }
        // You can do some transformation here and set fields on the event
        //if(key == "MySource")
        //    event.source = jsonObject[key];
        additionalInfo[newKey] = jsonObject[key];
    }
}
try
{
    gs.info("TransformEvents_generic received body:" + body);
    var jsonObject = JSON.parse(body);
    var event = new GlideRecord('em_event');
    event.source = "GenericJson"; //TODO: Need to define
    event.event_class = "GenericJsonClass"; //TODO: Need to define
    event.severity = "5";

    var additionalInfo = {};
    updateAdditionalInfo(event, "",jsonObject,additionalInfo);
    /*Iterates over Additional information JSON object and adds all nested objects' fields as fields of the Additional information object*/
    var notDone = true;
    while (notDone) {
        notDone = false;
        for (var key in additionalInfo) {
            if (Object.prototype.toString.call(additionalInfo[key])
                == '[object Object]') {
                notDone = true;
                updateAdditionalInfo(event,
                key,additionalInfo[key],additionalInfo);
                additionalInfo[key] = "";
            }
        }
    }
    gs.info("TransformEvents_generic generated additional information:" +
    JSON.stringify(additionalInfo));
    event.additional_info = JSON.stringify(additionalInfo);
    event.insert();
}
catch(er){
    © 2020 ServiceNow, Inc. All rights reserved.
    ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
}
Integrate AWS platform as a data source
Integrate Amazon Web Services (AWS) with Event Management. To add AWS platform as a data source, configuration is required in the AWS platform.

Role required: evt_mgmt_admin

When an AWS platform alarm arrives, Event Management:

- Extracts information from the original AWS platform alarm to populate required event fields and inserts the event into the database.
- Captures the content in the additional_info field.

The AWS platform transform script is located in Event Management > Event Listener (Push) > Listener Transform Scripts. In the Listener Transform Scripts page, click AWS Events Transform Script.

**Note:** The AWS transform script that is provided in the base system handles AWS CloudWatch alarms only. To handle Simple Notification Service (SNS) alarms that are other than AWS CloudWatch, create a new script or customise the AWS transform script.

1. In the AWS platform console, select Simple Notification Service. If an SNS topic does not exist, create a new one.
2. Under the SNS topic, create a new subscription.
   a) Take Topic ARN from the topic that you created. The Amazon Resource Name (ARN) is necessary for binding an Event Management alert to a CI.
   b) Set Protocol to: https.
   c) Set Endpoint to: https://<username>:<password>@<instance-name>.service-now.com/api/global/em/inbound_event?source=AWS.

   If AWS platform Multi-Factor Authentication (MFA) is enabled, when signing in to the AWS platform website, the user name and password are prompted for, as well as an authentication code from the AWS platform MFA device of the user.
3. Wait until the subscription changes from Pending to Confirmed and the subscription ARN is populated.
4. Create alarms in AWS platform to send to Event Management. Link the alarms to the SNS topic that you created.

These event rules are provided with the base system:
Integrate Azure platform as a data source
Integrate Microsoft Azure with Event Management. To add the Azure platform as a data source, configuration is required in the Azure platform.

Role required: evt_mgmt_admin and web_service_admin

The Event Management integration with Azure supports the Azure Classic Metric Alert format, also known as Insights Alerts. For this format, Event Management provides a dedicated listener, **Azure Events Transform Script**, see Dedicated listener. Several event rules for this format are provided with the base system, see Azure event rules provided with the base system. For information about how to receive events from other Azure formats, see Receive event from other Azure formats.

Activate the inbound event azure endpoint to enable receiving Azure platform alerts in Event Management, which works without security authentication:

1. Navigate to System Web Services > Scripted Web Services > Scripted Rest APIs.
2. Locate and click the Inbound Event script.
3. In the Resources area, click inbound event azure.
4. Select Active and then click Update.

**Dedicated listener**
Configure a dedicated listener that supports the Azure Classic Metric Alert format, as follows:

1. Open the Azure platform transform script, navigate to Event Management > Event Listener (Push) > Listener Transform Scripts.
2. In the Listener Transform Scripts page, click Azure Events Transform Script.

You can select to send Azure alerts either through the instance or the MID Server.

**Azure event rules**
Azure event rules provided with the base system are:

<table>
<thead>
<tr>
<th>Event rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azure host binding</td>
<td>Bind Azure platform alerts, on either the host or VM, to the host Hardware CI.</td>
</tr>
<tr>
<td>Azure vm binding</td>
<td>Bind Azure platform alerts, on either the host or VM, to the Virtual Machine Instance CI. To enable this rule, first disable the Azure platform host binding rule.</td>
</tr>
<tr>
<td>Azure MSSQL binding</td>
<td>Bind Azure platform MSSQL alerts to the Cloud database CI.</td>
</tr>
<tr>
<td>Azure database binding</td>
<td>Bind Azure platform database alerts, other than MSSQL, for example, NoSQL and Redis, to the Cloud database CI.</td>
</tr>
<tr>
<td>Azure WS binding</td>
<td>Bind Azure platform Web Server alerts to Cloud WebServer CI.</td>
</tr>
</tbody>
</table>

When an Azure platform alert message arrives, Event Management:
- Extracts information from the original Azure platform alert message to populate required event fields and inserts the event into the database.
- Captures specified content in the additional_info field.

1. In the Azure platform portal, create alert rules using the Alerts (Preview) interface. The definition of an alert rule in Azure platform portal has these parts:
   - Target: Specific Azure platform resource that is to be monitored.
   - Criteria: Specific condition or logic that, when seen in Signal, should trigger action.
   - Action: Specific call sent to a receiver of a notification - email, SMS, Webhook, and so on.

2. In the Webhook column, specify the endpoint URL in the format: https://<instance-name>.service-now.com/api/global/em/inbound_event_azure. For example:

   ```
   Webhook
   
   https://mist2.service-now.com/api/global/em/inbound_event_azure
   ```

**Receive events from other Azure formats** Event Management can receive events from other Azure formats, such as Azure Activity Alert (also known as audit log), and Azure log Alert (also known as unified log). Use this generic JSON target URL to collect events from other Azure formats:https://<<INSTANCE>>/api/global/em/inbound_event?source=genericJson. This generic URL can be used as-is, and requires an event rule to be configured to populate the correct fields in the alert.

Example of the **Transform and Compose Alert Output** section of an event rule to show the configuration to receive an alert when receiving alert rules from Azure in the Azure Activity Alert format.
Integrate Panopta as a data source

Integrate the Panopta cloud-based monitoring solution with Event Management. To add Panopta as a data source, configuration is required in Panopta.
Configure Panopta

To prepare Panopta as a datasource, follow the procedure described in this link: How do I integrate with ServiceNow.

When the Event Management integration is configured in Panopta, add the integration to any of your Panopta Alert Timelines so that the ServiceNow instance receives incident events accordingly.

Optional Event Management configuration

Configure an alert action rule for launch in context of Panopta. For more information, see Launch web application from alert and Create or edit an alert action rule.

Integrate Datadog as a data source

To add Datadog as a data source, configuration is required, both in Event Management and in Datadog.

The configuration tasks to collect events and incidents from Datadog includes:

Event Management configuration tasks

Datadog components are made available to Event Management users by selecting menu items. The menu functions are imported into Event Management through an update set. Perform these tasks in Event Management:

- Add the Datadog menu components.
- Configure the transform tables.

Datadog configuration tasks

In Datadog, configure the integration with Event Management and specify the required target interim table. Perform these tasks in Datadog:

- Configure the Event Management integration.
- Add the required credentials.
- Add the details for the target interim table.
- Test the integration.

Configure Event Management to add Datadog as a data source

Integrate Datadog with Event Management to collect event and incidents.

Role required: evt_mgmt_admin

Configure the Event Management environment for the collection of events or incidents from Datadog by importing the required XML Update Set. Enable the Datadog functions to be installed and then selected from the Event Management navigation pane.

1. In your local hard drive, create a temporary directory.
2. Import the DataDog-ServiceNow integration files into the temporary directory from GitHub https://github.com/DataDog/Datadog-Servicenow
3. Create an update set from the files in the temporary directory.
   For information about the procedure to create an update set, see Create a data source.
5. Under Related Links, click Import Update Set from XML.
6. In the **Import XML** form, select the Datadog update set XML file that you created in step 3, and click **Upload**.

7. In the **Retrieved Update Sets** form, click **Datadog**.

8. Click **Preview Update Set**. Ensure that there are no errors, then click **Commit Update Set**.

9. Grant yourself the `x_datad_datadog.user` user role.
   a) Navigate to **System Security > Users and Groups > Roles**.
   b) Search for `*datadog`.

10. In the search field of the **Application Navigation** pane, enter `datadog`. The Datadog menu items appear.
• Configure the transform tables.
• Perform the Datadog configuration tasks.

*Create a custom push connector*
You can create a customized listener to push event information to the instance or MID Server.

Customize script to enable Event Management to receive external events. By using a script to collect events, no further configuration is required. All other methods of collecting events require configuration.

• To collect event messages in the instance via web service API, see Event collection to the instance using web service API.
• To collect event messages using a script, see Event collection using script.

*Configure event collection for SNMP traps*
The SNMP listener runs on the MID Server, which acts as a collection endpoint for SNMP traps. The MID Server sends the traps to the ServiceNow instance for further processing as an event by Event Management.

Role required: evt_mgmt_operator

Ensure that the MID Server is assigned with either EventManagement or ALL applications.
Use the **MID SNMP Trap Listener** to listen for SNMP traps. In this configuration, the specified MID Server acts as a collection endpoint for SNMP traps.

1. Navigate to **Event Management > Event Listener (Push) > MID SNMP Trap Listener**.
2. Click **New**.
3. Complete the SNMP Trap Collector Context form:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name for the SNMP Trap Collector.</td>
</tr>
<tr>
<td>Short description</td>
<td>Short description of the MID Server extension execution context.</td>
</tr>
<tr>
<td>SNMP version</td>
<td>Specify the SNMP version:</td>
</tr>
<tr>
<td></td>
<td>- v1 and v2c</td>
</tr>
<tr>
<td></td>
<td>- v3</td>
</tr>
</tbody>
</table>

**Minimum Security Level**

The valid security level values are:

- None - No security level value set.
- noAuthnoPriv - Communication without authentication and privacy
- authNoPriv - Communication with authentication and without privacy.
- authPriv - Communication with authentication and privacy.

**Note:** This field appears only when v3 is selected in the **SNMP version** field.

<table>
<thead>
<tr>
<th>UDP Port</th>
<th>UDP port for SNMP Trap Collector to receive SNMP Traps. The default value is 1,162. However, if the port is already in use, you can use any other free port instead.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Status</th>
<th>The status of the trap listener. This field is blank until the listener is run. After the listener has run, the value is automatically populated to one of these statuses:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>- <strong>Started</strong>: The listener is running.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Stopped</strong>: The listener is not running.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Offline</strong>: The MID Server is down.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error</strong>: The listener failed with an error.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Execute on</th>
<th>The MID Server location for running this listener. Select from:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>- Specific MID Server</td>
</tr>
<tr>
<td></td>
<td>- Specific MID Server Cluster</td>
</tr>
</tbody>
</table>

The recommended setting is **Specific MID Server**.

Network hardware typically has to be configured to send to a specific IP address. For example, if the listener is moved to a different MID Server in the cluster, the trap would not be received.
ServiceNow    New York    IT Operations Management

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MID Server</td>
<td>Specify:</td>
</tr>
<tr>
<td></td>
<td>- The name of the designated MID Server when you select Specific MID Server in the Execute on field.</td>
</tr>
<tr>
<td></td>
<td>- The name of the designated MID Server cluster when you select Specific MID Server Cluster in the Execute on field.</td>
</tr>
<tr>
<td></td>
<td>If you selected Specific MID Server Cluster, a ServiceNow algorithm determines which server in the cluster runs the listener.</td>
</tr>
</tbody>
</table>

| Executing on        | The name of the MID Server on which the extension is running. This field shows the name of the MID Server even if the MID Server is down. If you stop the extension, this field is empty. |
|                     | This field is automatically populated.                                                                                                       |

4. Right-click the form header and select Save.
5. In the Credentials for SNMPs section, specify the required SNMP v3 credentials that the SNMP listener uses for authentication.

**Note:** The Credentials for SNMPs section appears only when v3 is selected in the SNMP version field.

6. In the Related Links section, click Test parameters.
7. Click Update.
8. Optional: If the MID Server fails to start, review the MID Server log errors.

In Related Links, these actions may be run against the SNMP traps listener:

**Commands available in the SNMP trap listener form**

<table>
<thead>
<tr>
<th>Related Link</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>If it is currently not running, starts the listener. The listener is started on the configured MID Server and port number.</td>
</tr>
<tr>
<td>Stop</td>
<td>Stops the running listener on the configured MID Server. If the listener is not running, no action is taken.</td>
</tr>
<tr>
<td>Restart</td>
<td>Stops, then starts the listener on the configured MID Server and port number.</td>
</tr>
<tr>
<td>Test parameters</td>
<td>Tests that the port is a valid number.</td>
</tr>
<tr>
<td>Update parameters</td>
<td>If you make changes to the listener configuration, use this option to update the parameters of the currently running listener. First, the parameters are tested for validity. If the parameters are valid, the listener disconnects and reconnects with the new parameters.</td>
</tr>
</tbody>
</table>

**Note:** Updates to the SNMP trap listener take effect only after Restart or Update parameters are clicked.

Configure the SNMP traps listener to receive OEM traps
Configure the SNMP trap listener to receive traps from Oracle Enterprise Manager (OEM).
Role required: evt_mgmt_admin

Event Management can process Oracle Enterprise Manager Cloud Control SNMP traps as events. For the SNMP trap listener to receive traps from OEM, you must designate the MID Server that runs the SNMP trap listener as a recipient of the trap. Upon receiving an OEM trap, the MID Server sends the trap to the ServiceNow instance for further processing as an event by Event Management.

The OEM 12c Trap event rule is provided with the default base instance, as well as these alert action rules:
- Oracle EM Launch Target Status
- Oracle EM Launch View Event

1. Navigate to **Event Management > Event Listener (Push) > MID SNMP Traps Listener**.
2. Click **New**.
3. Fill in the fields, as described in **Configure event collection for SNMP traps**. For the **SNMP version** field, select: v1 and v2c.
4. Click **Submit**.

In Oracle Enterprise Manager Cloud Control, configure the MID Server as a trap listener target.

Configure SNMP Trap collection for high availability
For SNMP traps, the MID Server requires failover configuration for the trap listener.

You can configure two MID Servers for failover. Because the SNMP Trap listener on the MID Server receives inbound traffic on the IP address and port of the MID Server, you configure each MID Server to receive the same SNMP traps. Only one MID Server is active at any point in time, therefore Event Management does not receive duplicate traps.

**Configure event collection from email**
Configure an inbound email action to send email notifications when events and alerts are triggered.

Role required: evt_mgmt_admin

When you add an inbound email action to send email, Event Management uses inbound email actions to process inbound email in the same manner as any external event. The email content is used to generate and update events, use rules to generate alerts, or change event severity. The severity is updated after the impact calculation and stored in the em_impact_status table.

1. Navigate to **System Properties > Email Properties**.
2. In the **Inbound Email Configuration** section, in the **Email receiving enabled** option, select **Yes** to enable the collection of events from email.
3. To configure the email information to pass to Event Management for event and alert processing, navigate to **System Policy > Email > Inbound Actions** and in the list of Inbound Email Actions records search for and open the default **create event** form.

   a) In the **Inbound Email Actions** form, specify these values.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Select the check box to enable the email notification.</td>
</tr>
<tr>
<td>Description</td>
<td>Type a description for this email message.</td>
</tr>
<tr>
<td>Target table</td>
<td>Select <strong>Event (em_event)</strong>.</td>
</tr>
<tr>
<td>Action type</td>
<td>Select <strong>Record Action</strong>.</td>
</tr>
</tbody>
</table>

   b) Click the **Actions** tab and select these **Field Actions**.
For example, you can select **Subject** as the portion of the email to use.

4. To customize the inbound parameters and map the text to event variables, add a **Script**. For example:

```javascript
current.source = 'email';
current.event_class = 'email';
current.description = email.body_text;
current.time_of_event = new GlideDateTime();
current.insert();
```

5. From the instance, send an email that contains matching data, and then confirm that the event or alert information appears in Event Management. For more information sending email messages to create an instance, see **Inbound email actions**.

**Note:** Ensure that the user who sends the email has the role **evt_mgmt_integration**.

6. Click **Submit**.

The inbound email is sent to the em_event table and regular Event Management processes continue, for example, event rules and alert management rules run.

**Manage and monitor alerts**

An alert is a notification generated by Event Management for selected events that are considered to be important and require attention. The generation of alerts is based on event rules.

After alerts generate, the manner in which you can monitor and resolve alerts is based on alert configuration settings and properties, and alert impact calculation.

**Contextual search in alerts**

Use contextual search to search for and attach a knowledge base (KB) article from within an alert.

**Role required:** **evt_mgmt_user**

In an alert, search for related KB articles to provide additional information about the alert. You can preview the text of the article and optionally attach the article to the alert.

The search query is based on the contents of the short description of the alert. This search query comprises the following fields from the alert:

- Node
- Type
- Resource
- Description

When performing a search for related articles, a list of articles appears below the related search field. You can preview an article. If relevant, attach the KB article to the alert.

1. Navigate to **Event Management > All Alerts**.
2. In the Alerts list, select an alert.
3. In the alert form, click **Related Search Results**.
4. Review the list of suggested knowledge articles, and:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preview the article</td>
<td>Click Preview next to the article. After reviewing the article in the pop-up window, click Attach to Alert if it is relevant, or click (X) to close the article.</td>
</tr>
<tr>
<td>Attach the article without previewing it</td>
<td>Click Attach next to the article.</td>
</tr>
</tbody>
</table>

When you attach an article, a URL to the KB article is added to the Work Notes field of the alert. Save the alert to keep the changes.

If performing a search for related articles and the short description of the alert is blank and when you need to search according to your own keywords:

- You can enter your search query in the Related Search text box.
- Click Related Search Results.

5. Review the list of suggested knowledge articles, as in step 4.

6. Click Update.

**Alert management rules for resolving alerts**

Configure Event Management to automatically respond to alerts. Create alert management rules to determine the required response, which might be to open an incident, knowledge base, open any kind of task, launch remediation action, and so on.

Alert management rules that are provided with the base system are available to assist you to respond to alerts. Use filters to specify conditions. The alert management rule takes the remedial action that is specified when the conditions are met. For example, to launch the required subflow or open an incident based on an alert. The alert is automatically updated to show the execution history, showing what actions were taken.

Users with the evt_mgmt_admin role can use the alert management rule designer to create and configure alert management rules that can act on the specified alerts. Define rules with complex filters to determine which alert the rule must apply to. Create rules to launch applications, URLs, subflows, remediation actions, or take other actions, for example, open an incident. For more information, see [Create an alert management rule](#).

Users with the evt_mgmt_operator role can manually run alert management rules.

**Alert management rule workflow**

The workflow to create and run an alert management rule is described in the following steps.
1. Create an alert management rule to respond to the alerts that you specify.
2. Configure a filter for the rule, to ensure that the rule applies only to those alerts that match the filter.
3. Configure the actions to be taken to resolve the alert.

Components of alert management rules

The components of an alert management rule are:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert Info</td>
<td>Create the rule.</td>
</tr>
<tr>
<td>Alert Filter</td>
<td>Specify a filter to determine to which alerts this rule must be applied to. Related list conditions can be specified.</td>
</tr>
<tr>
<td>Actions</td>
<td>Specify the response to the alert, for example, run a subflow, perform remediation action, launch an application, or launch a URL in a browser.</td>
</tr>
</tbody>
</table>

How rules are applied to updated alerts

Alert management rules run on all open alerts that have been updated. Rules do not run on closed alerts, even if they have been updated. Whether the actions that are configured in the rule are applied to the alert, depends on the conditional filters of the rule. For example, assume that a rule has a condition that an email message is sent when the severity of an alert changes to Major. This rule applies to an alert updated by a severity change from Warning to Major.

Use of filters and other actions

Create a filter so that the rule does not act on every update of the alert, only when the condition that you set occurs. For example, an entry in the Work notes field of an open alert is an update, but you can configure the rule so that updates that are not relevant do not cause the rule to run. Configure alert management rules to run only when the update is significant to the rule. Another example is to set a condition so that the alert management rule runs only when the alert severity is Critical.

You can:

- Specify a filter to determine which alerts the rule applies to.
- In the Related List Conditions section, configure further conditions, for example, with an Alert > Parent relationship, to filter for any alerts that were received today.
- Respond to alerts. For example, by using subflows and workflows, create incidents for primary alerts with critical severity, or open a search engine in a browser to search for data according to the description field of the alert.
- Apply remediation. Remediation is based on Orchestration workflows that can be scripted to perform remediation tasks such as gathering system information or rebooting a server.
Scheduled job that checks alert management rules

Alert management rules are checked every 11 seconds by the default Event Management - Evaluate Alert Management Rules scheduled job. The job then executes the required actions.

Migrate existing alert action rules

Existing alert action rules from an earlier release can be migrated to become alert management rules. Only after migrating an alert action rule to an alert management rule, can the rule be modified. For more information, see Migrate an alert action rule to an alert management rule. Create an alert management rule

Create an alert management rule to track alerts and resolve them by determining the required response, for example, to open an incident, or launch remediation action.

To enable remediation with a subflow, you can use a subflow that is available with the base system, or you can create your own subflow. To create a custom subflow, navigate to Flow Designer > Designer. In the Flow Designer window, click Subflows. Click Alert Management Template. The template opens in Flow Designer in read-only format. Click the more actions icon and select Copy subflow. In the New Subflow Name field, enter the name of the new subflow. If you want to limit the application scope of the subflow, in the Application field select the required application. Click Copy. In the Flow Designer editor, you can specify values for the fields, add actions, flow logic, and subflows. For example, click Action, select Alert Management Content and then select Update Execution With Task. Continue by entering information in the required fields and then click Done. To save the subflow in draft status, click Save. To save and publish the subflow so that it can be used in the alert management rule designer, click Publish. For more information about using Flow Designer, see Create a subflow.

To enable remediation with a workflow, create the workflow to remediate alerts.

Navigate to Workflow > Workflow Editor. In the Welcome tab of the Workflow Editor, click New Workflow. In the Table field, select Remediation Task (em_remediation_task).

After you finish configuring the workflow, make sure that you publish it. For information about creating a workflow, see Create a workflow.

Role required: evt_mgmt_admin

Use alert management rules to track and resolve alerts.

While working in the alert management rule designer, you can work in multiple sections without losing information in any section.
Create alert management rules that:
- Locate other alert management rules that have relevance to the selected alert.
- Determine when the execution of the rule takes place.

You can configure alert management rules to:
- Automatically generate and link incidents, tasks, or knowledge articles to alerts.
- Automatically apply a remediation workflow or enable users to manually run remediation.
- Automatically construct a URL that is created according to the value of specified fields in the alert.

To assist you, several alert management rules are provided with the base system. You can use them as presented or you can use them as examples to build custom alert management rules.

### Alert management rules provided with the base system

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open sensor dashboard in PRTG</td>
<td>The sensor dashboard in the Paessler PRTG Network Monitor (PRTG) application opens.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| Oracle EM Launch Target Status and View Events | Launch Oracle Enterprise Manager to view:  
  - Target Status  
  - Event for alerts from source Oracle EM | Yes    |
| Drilldown to OMI                          | Drill down to the HP Operations Manager i (OMI) application.                 | Yes    |
| Create Incident on Primary Critical Alert | Create an incident for primary critical alerts. The incident can be created automatically or manually. | No     |
| Search Google for "description"          | Open Google Search in a browser to search for data according to the description that appears in the alert. | Yes    |
| Create Incident                          | Create an incident for all alerts that are not in maintenance state. The rule runs automatically on selective update. | No     |
| Create Incident Manually                  | Manually create an incident for alerts that are not in maintenance state.    | Yes    |
| Create Major Incident Candidate           | Create a major incident candidate for all alerts that are not in maintenance state and are not secondary alerts.  
  A major incident candidate can be promoted to become a major incident. | No     |
| Create Major Incident                     | Create a major incident for all alerts that are not in maintenance state and are not secondary alerts. | No     |

If your instance was upgraded from Kingston, the alert action rules that were provided with the Kingston base system are available to you. However, if you modified any of the rules, the changes made are not carried over.

1. **Navigate to** **Event Management > Rules > Alert Management.**
2. **Click** **New.**
3. On the form, fill in the fields.

**Alert Management Rule form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name for the rule.</td>
</tr>
<tr>
<td>Active</td>
<td>Check box for enabling the rule.</td>
</tr>
<tr>
<td></td>
<td>If this check box is selected, you must specify:</td>
</tr>
<tr>
<td></td>
<td>- in the Alert Filter section, an alert filter</td>
</tr>
<tr>
<td></td>
<td>- in the Actions section, at least one of any of these actions:</td>
</tr>
<tr>
<td></td>
<td>- active subflow</td>
</tr>
<tr>
<td></td>
<td>- workflow</td>
</tr>
<tr>
<td></td>
<td>- quick response</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which rules are evaluated when multiple rules are defined for the same alert. Alert management rules are evaluated in ascending order. The default value is 100.</td>
</tr>
<tr>
<td>Multiple alert rules</td>
<td>Instruction about whether to search for additional rules:</td>
</tr>
<tr>
<td></td>
<td>- Search for additional rules--execute the current rule then continue and execute other matching rules by the order of rule priority, where the lower number has the higher priority.</td>
</tr>
<tr>
<td></td>
<td>- Stop search for additional rules--execute only the current rule for the alert that matches the defined filter.</td>
</tr>
<tr>
<td>Description</td>
<td>Descriptive text to describe the rule.</td>
</tr>
</tbody>
</table>

4. Click **Alert Filter** and specify conditions for alerts that this rule is applied to.

5. On the form, fill in the fields.

**Alert Filter stage**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule is activated when</td>
<td>Rule execution takes place when:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Alert changes to filter</strong>—content changes to the alert cause the alert to match the filter. If the filter is matched on following update of the alert, the rule is not applied. If the alert was closed and then reopen, at the next update of the alert and the filter is matched, the rule is applied. Thereafter, when there is an update of the alert, the rule is no longer applied.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Alert matches filter</strong>—the content of the alert matches the filter. On following update of the alert and if the filter is matched, the rule runs and is applied to the alert. The rule remains applied for every matching update.</td>
</tr>
<tr>
<td>Alert filter</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Preview</td>
<td>Function to preview alerts that match the specified condition. A hyperlink shows how many alerts match the filter.</td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Preview" /> <strong>13 records match condition</strong></td>
</tr>
<tr>
<td></td>
<td>If you click the hyperlink, the browser opens another tab that lists alerts in the Alerts (em_alert_list) table. The list shows which alerts match the rule, including closed alerts. Alerts that have already been run by the rule are not marked in any way. You can click any alert to view further details.</td>
</tr>
<tr>
<td>Conditions</td>
<td>Conditions that, if fulfilled, cause the filter to be applied. For more information about building conditions, see <a href="#">Using the condition builder</a>.</td>
</tr>
<tr>
<td></td>
<td>To add another condition, click <strong>New Criteria</strong>.</td>
</tr>
</tbody>
</table>
| Related List Conditions | Conditions to include a relationship with another table in the filter.  
1. Click **Table** and select the required table.  
2. Specify the conditions for this filter.  
For more information about creating related lists, see [Add related list conditions](#).                                               |

### 6. Click Actions.

In this section, you can configure these action types as a response to alerts or to remediate alerts:

- Remediation Subflows—Execute a subflow provided with the base system.
- Remediation Workflows—Execute a workflow that you previously published.
- Launch Applications—Open applications and browsers that you configure.

### 7. Optional: In the Remediation Subflows area, to add subflows:

a) Under **Subflow**, double-click the cell.

b) Click the search icon ![search icon](image).  
The list of subflows provided with the base system appears. For more information, see [Event Management subflows provided](#).  

c) From the subflow list, select a subflow.

d) Repeat, adding as many subflows as required.
e) If you want to specify when the subflow must be executed, under **Execution**, double-click the cell.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic</td>
<td>The subflow is executed automatically when the rule is matched.</td>
</tr>
<tr>
<td>Manual</td>
<td>Execute the subflow if required when the rule is matched.</td>
</tr>
<tr>
<td>Both</td>
<td>When the rule is matched, the subflow is executed automatically and you can optionally execute the subflow again manually.</td>
</tr>
</tbody>
</table>

f) Under **Automatic executions limit**, double-click the cell and enter the required integer. Use this field to configure how many times the subflow is executed.

g) If you want to enable the subflow to be executed, under **Active**, double-click the cell and select `true`.

A link in the cell under **Link to Flow Designer** appears only after a subflow has been selected and the rule has been saved.

### Link to Flow Designer

`/flow-designer.do?sysparm_nostack=true#/sub-flow-designer/b1e21924937b0300415c74`
### Workflow execution options

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic</td>
<td>The subflow is executed when the rule is matched.</td>
</tr>
<tr>
<td>Manual</td>
<td>When to execute the subflow when the rule is matched. For example, you can run the remediation from the <strong>Quick Response</strong> option. For more information, see <a href="#">Apply a quick response in an alert</a>.</td>
</tr>
<tr>
<td>Both</td>
<td>When the rule is matched, the subflow is executed automatically and you can optionally execute the subflow again manually.</td>
</tr>
</tbody>
</table>

- **e)** Optional: Double-click the cell under **Automatic executions limit** and enter the required integer.
  
  Use the **Automatic executions limit** field to configure how many times the workflow is executed. This value sets the maximum number of executions of the workflow as long as the alert is open. The counter is reset when the alert is closed.

- **f)** Optional: To enable the workflow to be executed, double-click the cell under **Active** and select true.
  
  For information about how to create a workflow, see [Create a workflow](#).

9. To add instructions to launch applications or to open browser windows, in the **Launch Applications** area:

   Any URL-based action can utilize the alert parameters and the URLs can refer to wikis, messaging services, REST APIs, and so on.

   - **a)** Under **Display Name**, double-click the cell.
     
     Specify a name for the link.

   - **b)** In the **URL** field, compose the URL using data from the alert in the format: `http://{source}.com:{port}/${cmdb_ci.name}

     The **Active** field is automatically updated.

10. Click **Submit**.

The alert management rule that was created is added to the list of available rules that can be used to resolve alerts.

**Event Management subflows provided**

In the Remediation Subflows area of alert management rules, the subflows provided with the base system are available.

**Subflows available**

Navigate to **Event Management > Rules > Alert Management** and click **New**. Click the **Actions** tab. In the Remediation Subflows area, double-click the **Insert a new row** field.
Click the search icon to add subflows. The list of subflows that are provided with the base system appears.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledge Alert</td>
<td>Subflow to mark the alert as being Acknowledged. Acknowledge an alert to show that further attention is required.</td>
</tr>
<tr>
<td>Attach Knowledge Article (legacy)</td>
<td>Subflow to attach a knowledge article to the alert. This subflow is provided for instances that are migrated from legacy releases (prior to the London release).</td>
</tr>
<tr>
<td>Change Alert to Maintenance Mode</td>
<td>Subflow to mark the alert as being in Maintenance.</td>
</tr>
<tr>
<td>Close Alert</td>
<td>Subflow to mark the alert as being Closed.</td>
</tr>
<tr>
<td>Create Incident</td>
<td>Subflow to create an incident. Fields from the alert are used to populate the matching fields in the incident that is created.</td>
</tr>
<tr>
<td>Create Major Incident Candidate</td>
<td>Subflow to create a major incident candidate. Fields from the alert populate the matching fields in the major incident candidate that is created. A major incident candidate can be upgraded to become a major incident.</td>
</tr>
</tbody>
</table>

**Note:** Add the Knowledge article column to the Alert Management Rules (em_alert_management_rule) table, and select an article to attach to an alert when the rule executes.

**Note:**
- If there is an existing incident that is attached to the alert, this subflow is not activated.
- If the alert is in Maintenance, an incident is not created.

**Note:**
- If there is an existing incident that is attached to the alert, this subflow is not activated.
- If the alert is in Maintenance, a major incident candidate is not created.
- If the Role in group is Secondary, the major incident candidate is not created.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create Major Incident from Alert</td>
<td>Subflow to create a major incident from alert. Fields from the alert are used to populate the matching fields in the major incident that is created. Note: If there is an existing incident that is attached to the alert, this subflow is not activated. If the alert is in Maintenance, an incident is not created. If the Role in group is Secondary, the major incident candidate is not created.</td>
</tr>
<tr>
<td>Create Task (legacy)</td>
<td>This subflow uses a task template, if provided, or the EventMgmtCustomIncidentPopulator script for instances migrated from legacy releases (prior to the London release). If configured, apply the task template. Note: Add the Task template column to the Alert Management Rules (em_alert_management_rule) table, and select a task template and task to apply when the rule executes.</td>
</tr>
<tr>
<td>Overwrite Alert Template (legacy)</td>
<td>This subflow applies the alert template. This subflow is provided for instances that are migrated from legacy releases (prior to the London release). Note: Add the Task type column to the Alert Management Rules (em_alert_management_rule) table, and select an alert template to apply when the rule executes.</td>
</tr>
</tbody>
</table>

1. Select the subflow that you need.
2. To customize a subflow, see Create a custom subflow. This topic also describes the input parameters in a subflow.
3. To specify when the workflow must be executed, double-click the cell under Execution.
Create a custom subflow
You can create a subflow according to your requirements. You can then use the custom subflow to resolve alerts.

Role required: evt_mgmt_admin

Copy an existing subflow, and then customize it. Publish the custom subflow so that it is available to be used to resolve any alert. For more information, see Create a subflow.

Note: If your ServiceNow instance uses domain separation, you must ensure that the new custom subflow that you created operates only on the data of the designated domain.

For the purposes of explanation, the Create Incident subflow is used. For more information about how to customize a subflow, see this Knowledge Base article KB0722376.

1. Navigate to Flow Designer > Designer.
2. In the Flow Designer window, click Subflows.
3. Click Create Incident.

The Create Incident subflow opens in Flow Designer in read-only format. The inputs in the Create Incident subflow are described in the following table.

Create Incident input parameters

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alertRuleName</td>
<td>The name of the alert management rule that specifies this subflow, for example IDD.</td>
</tr>
<tr>
<td>alertRuleId</td>
<td>The sys_id of the alert management rule, for example, 1ae122e84f5f1300b28392918110c71e.</td>
</tr>
<tr>
<td>alertGR</td>
<td>The alert record that matches the filter that is defined in the alert management rule, for example, Alert0027931</td>
</tr>
<tr>
<td>executionId</td>
<td>The ID of the execution to be retrieved, for example, 000182ebdb66630091187f1daf9619d4</td>
</tr>
<tr>
<td>userName</td>
<td>The current user’s name, for example, System Administrator</td>
</tr>
<tr>
<td>userDisplayName</td>
<td>The name that the current user specified to be displayed, for example, admin</td>
</tr>
</tbody>
</table>

For example, in an alert management rule that has processed alerts, in the Alert Executions area, click the Link to execution of the required alert.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
The Flow Designer opens. Click **Subflow Inputs and Outputs**. The input variable names of the subflow are displayed with their respective values.

![Subflow Inputs and Outputs](image)

<table>
<thead>
<tr>
<th>VARIABLE NAME</th>
<th>TYPE</th>
<th>RUNTIME VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>alertGR</td>
<td>Reference</td>
<td>Alert0027931</td>
</tr>
<tr>
<td>alertRuleId</td>
<td>String</td>
<td>1ae122e84f5f13</td>
</tr>
<tr>
<td>alertRuleName</td>
<td>String</td>
<td>IDD</td>
</tr>
<tr>
<td>executionId</td>
<td>String</td>
<td>000182ebdb666</td>
</tr>
<tr>
<td>userDispName</td>
<td>String</td>
<td>System Admin</td>
</tr>
<tr>
<td>userName</td>
<td>String</td>
<td>admin</td>
</tr>
</tbody>
</table>

4. In the subflow header, click more actions and select **Copy subflow**.

5. In the **New Subflow Name** field, enter the name of the new subflow.

6. To limit the application scope of the subflow, in the **Application** field, select the required application, otherwise select **Global**.

7. Click **Copy**.

The new subflow is created and is opened as a tab in the subflow designer editor. The inputs provided by the template appear.

In the editor, you can delete subflow inputs or outputs, as required.

8. To delete an input, click ![Trash Can](image).

9. To add an input, click ![Plus Sign](image).

A variable is added to the custom subflow.

a) Double-click the variable to give it a recognizable name, for example, `alertDescription`.

b) In the **Type** column, select the type variable that is being defined, for example, `String`.

c) To make the input a mandatory configuration option, select the **Mandatory** flag.

d) Click ![Down Arrow](image) to view the advanced options and define values.
Inputs provide advanced options based on their data type. All inputs have advanced options to add a hint or provide a default value. Use advanced options to guide flow designers through adding and configuring a subflow to a flow. For example, create a choice input to provide flow designers with a pre-defined list of configuration options to choose from. For more information about the configuration options available to particular data types, see field types.

10. Create subflow outputs by defining the names and data types.
   a) Click \( + \) to add output.
   b) Define the name and the data type.

11. To save the subflow in draft status, click Save.
12. To save and publish the subflow so that it can be used in the alert management rule designer, click Publish.

Include the modified subflow in an alert management rule to resolve alerts according to your customized requirements.

**Alert Executions Information**
Alert executions information provides a reference to the alert management rule actions that are performed. This information appears in the Alert Management Rule record only if an alert matches the filter in the rule and an action was performed. You can click any link in the Alert Executions list to open the referenced item.

**Alert Executions**
In the Alert Management Rule record, the Alert Executions list shows key related records for the rule. Among the

### Alert Executions list

- Alert0010009 Create Incident
  - \( /\text{flow-designer.do?sysparm_nostack=true#/operations/...} \)
- Alert0010008 Create Incident
  - \( /\text{flow-designer.do?sysparm_nostack=true#/operations/...} \)
Information in the Alert Executions list

The columns in the Alert Executions list are populated with details of the alerts that matched the alert management rule and the actions that were run. The alerts shown are the alerts that match the filter in the rule.

View alert execution information
You can click any link in the Alert Executions list to view the alert execution information of the referenced item. This information appears in the Alert Management Rule record only after an alert matched the filter in the rule and an action was performed.

Role required: evt_mgmt_admin
Among the information shown in the Alert Executions list is:

- The list of alerts that matched the filter of the rule.
- The tasks, including incidents, that were opened.
- Which remediation workflows ran and which subflows ran.

2. In the Alert Management Rules list, click the required rule.

If an alert matched the filter in the rule and an action was performed, the Alert Executions list displays in the Alert Management Rules record. This list is displayed irrespective of which stage of the alert management rule has been selected.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert</td>
<td>Alert number of the alerts that matched this alert management rule.</td>
</tr>
<tr>
<td>Action name</td>
<td>Name of the action that was taken to resolve the alert (subflow or workflow).</td>
</tr>
<tr>
<td>Link to execution</td>
<td>Link to the action execution log that was created when the action was performed.</td>
</tr>
<tr>
<td>Related task</td>
<td>Link to the task that was created when the alert was resolved. For example, if the action for a rule is to create an incident, this field provides the incident number with a link to open the incident. A link is provided to the workflow remediation task that is created to monitor the remediation.</td>
</tr>
<tr>
<td>Log</td>
<td>Notice or information showing the result of the execution action, for example, how executions took place, &quot;1 out of 2 executions.&quot; This information does not indicate that the subflow was executed successfully, rather that the call to that subflow was successful and that the subflow is correctly configured.</td>
</tr>
<tr>
<td>Automatic run</td>
<td>Indication of how the action ran: automatically (true) or manually (false).</td>
</tr>
<tr>
<td>Created</td>
<td>Date and time that the entry was created.</td>
</tr>
</tbody>
</table>
Migrate an alert action rule to an alert management rule

Existing alert action rules from an earlier release can be executed, but cannot be modified. Alert action rules that have been migrated become alert management rules and all the definitions of the alert action rules are migrated to the alert management rule format. Migrated rules can be modified.

Role required: evt_mgmt_admin

Existing alert action rules are active, read-only rules. An alert action rule can be modified only after it is migrated to an alert management rule.

2. Select an alert action rule to migrate or delete.

3. Click Migrate.
   
   If you clicked Migrate, the list of migrated rules can be found, together with new alert management rules, by navigating to Event Management > Rules > Alert Management.

   After a successful migration, all the definitions of the alert action rules are migrated to the alert management rule format.

   When an alert action rule has been migrated successfully, the alert action rule can no longer be modified. However, the new alert management rule can be modified.

   If the alert action rule that was migrated had a task template, you can customize the alert management rule designer to display the task template name.

   1. In the header of the alert management rule, click Additional actions.
2. Select Configure > Form Layout.
3. In the **Available** pane, select **Task template** and move it to the **Selected** pane. Position it as required.

4. In the **Task template** field, select the required template.

5. In the Actions tab, in the Remediation Subflows area, select the Create Task (legacy) subflow to run the task template.

---

### Alert lifecycle configuration

Event Management provides various modules, templates, and properties for configuring alerts and the actions that execute for these alerts.

**Configure the alert active interval**

The active interval property determines how Event Management handles a new event that is similar to events that appear on an existing alert. Based on the active interval, event, and existing alert information, the event information is added to either the existing alert or a new alert.

Role required: evt_mgmt_admin

When the new event is received, Event Management compares the active interval time to the elapsed time between the **Time of event** and **Initial event time** field values on the alert. Then either the existing alert is updated or a new alert is created, as explained in the following table.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>The new event occurs before the active interval elapses for the existing alert.</td>
<td>The alert is updated with the event information. The event is identified as a recurrence of an existing issue and its severity is copied to the alert. If the alert was closed before the active interval elapses, the alert is reopened and updated with the event information.</td>
</tr>
<tr>
<td>The new event occurs after the active interval elapses for the existing alert.</td>
<td>A new alert is created because the event is identified as a new issue. The event information is added to the new alert.</td>
</tr>
</tbody>
</table>

Change the active interval by editing the **Active interval (in seconds)**, within which a new event **reopens a closed alert** property value.

1. Navigate to **Event Management > Settings > Properties**.
2. Change the number of seconds for the **Active interval (in seconds)**, within which a new event reopening a closed alert property.

3. Click **Save**.

**Configure alert flapping**

You can set flapping properties to determine when an alert enters and exits the flapping state.

Role required: evt_mgmt_admin

Flapping condition that occurs when the event source continues to generate events even after its associated alert has been closed. Flapping causes the status of the resource to repeatedly fluctuate between an **Info** Severity and another severity that requires attention, for example, **Critical**.

An alert enters the flapping state when its current **Flap Count** value reaches or exceeds the given **evt_mgmt.flap_frequency** property value within the time period specified by the **evt_mgmt.flap_interval** property.

An alert exits the flapping state when the time interval between the latest occurrence of the event and the **Flap last update time** of the alert reaches or exceeds the **evt_mgmt.flap_quiet_interval** property value. An alert can also exit flapping in these conditions:

- When an incoming event arrives after the **flap_quiet_interval** time has elapsed. The new alert generates and its state value depends on the incoming event severity.
- After a background job, which runs every five minutes, verifies that no alert updates occurred and the **flap_quiet_interval** time has elapsed. If there are no updates, the alert shows the previous state value that it had before the background job ran.

1. Navigate to **Event Management > Settings > Properties**.

2. Edit the following properties, as appropriate.
   
   - **Minimum time in seconds before updating an alert for identical events**
     (**evt_mgmt.update_alert_restricted_fields_elapsed_time**)
   - **Flap interval (in seconds), within which an alert enters the flapping state**
     (**evt_mgmt.flap_interval**)
   - **Flap frequency, frequency an alert must reoccur to enter the flapping state**
     (**evt_mgmt.flap_frequency**)
   - **Flap quiet interval (in seconds), quiet time that must pass for an alert to exit the flapping state**
     (**evt_mgmt.flap_quiet_interval**)

3. Click **Save**.

You can monitor the **Event Management > All Alerts** list for alerts that are in the flapping state. For details, see [View alerts in the flapping state](#).

**Alert impact calculation**

Impact calculation shows the magnitude of an outage on CIs, services, alerts, and alert groups. As an alert generates, factors such as impact rules and CI relationships are used to calculate the severity. The alert severity is then displayed on the impact tree, application service maps, and dashboards.

Impact calculations are available for business services, application services, and alert groups. The following factors are used to calculate the overall impact of an outage.

- **Impact rules**.
- **Number of related active alerts**.
- **Past history of the affected CI**.
- **Relationships between CIs for a particular business service or application service**.

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
- If the CI element includes a network or storage devices.
- If the CI is in maintenance. When CIs are in maintenance, alerts on the CI are excluded from impact calculation.

**Note:** CIs are considered to be in maintenance not only when an active change request is scheduled, but also when the Status field of the CI is set to **In Maintenance**. In both cases, the CI is excluded from impact calculation and alerts for the CI do not appear in the Alert console.

**Note:** When a child CI is put in maintenance, it also places the parent CI in maintenance.

If there is a connection between services, the impact of one service on the other is also calculated.

Impact calculations use information from various sources to set the alert severity.
How impact is calculated

Impact calculation varies depending on the CI relationships for a business service or application service. Additional factors, such as change requests, network paths, storage paths, and related CIs all affect impact calculation.

You can learn about the impact tree from the following video tutorial.

Business services

The following impact calculation flow operates for alerts where the outage does not affect a network or network storage. Event Management performs the following steps:

1. Create a business service map. Use the Service Configuration Item Associations (svc_ci_assoc) and CI Relationships (cmdb_rel_ci) tables to create child-parent relationships in the business service or application service.

2. If there is no CMDB path from the business service to the CI but an association appears in the svc_ci_assoc table, show a Depends-on relationship between the business service and the CI. Otherwise, show no connection.

3. For an application service, if the CIs assigned to the service are also connected to the service in the CMDB, the map keeps the hierarchy between CIs as they appear in CMDB. The CI service assignments appear in the Service Configuration Item Associations section of the Application service form. If there is no connection to the service in the CMDB, the CIs appear directly under the application service in the map.

4. Create the impact tree. Mark the magnitude of an outage by 100% down, 60% affected, 40% impaired, or 20% impaired. If the items in two or more clusters are affected, the impact is 100% down.

Change requests and the In Maintenance status

If an active change request is scheduled for the CI or if the Status of the CI is In Maintenance, all alerts on the affected CI are excluded from impact calculation. The Alerts tab and Alert console also temporarily hide all corresponding alerts. The impact tree shows the CI in green with a note of (In Maintenance). The impact tree and the business service map temporarily show CIs in green.

Note: CIs are considered to be in maintenance not only when an active change request is scheduled, but also when the Status field of the CI is set to In Maintenance. In both cases, the CI is excluded from impact calculation and alerts for the CI do not appear in the Alert console.

For a business service, all alerts on CIs in the business service are also hidden from the Alerts tab. The entire business service is shown in green on the impact tree. For a host with an active change request, the host applications are considered as one unit. All child applications are treated in the same manner as the host until the change request is no longer active. For additional information, see How alerts work with CIs in maintenance.

Network paths

To account for network redundancy, Event Management uses a separate impact calculation. You can see network topology or path changes in the business service. The following impact calculation flow operates for alerts where a network path is affected. Event Management performs the following steps:

1. Create a business service map for the affected network.
- Use the host ID and target IP information from the alert and the network path from the Network Paths (sa_network_paths) table.
- Use the elements in the network path that derive from the Configuration Item (cmdb_ci) table. Also, use the elements that are associated to the path, from the Infra Path To Elements (sa_infra_path_assoc) table.
- Set the relationships. The application CI has a Depends on::Used by relationship on an element in the path that is defined in the CI Relationship (cmdb_rel_ci) table. In the relationship, the application CI is the parent and the element in the network path is the child.

2. Calculate a separate severity for each regular element in the path. Each regular element in the path contributes its own severity to its ancestors up to the application CI where the path originated from.

3. Calculate all redundant elements in the path with the redundancy rule by reducing the severity on the impacted CIs by one level. For example, if the severity is Critical, the redundancy rule decreases severity by one level to Major.

4. Create the impact tree. Mark the magnitude of an outage by 100% down, 60% affected, 40% impaired, or 20% impaired. If the items in two or more clusters are affected, the impact is 100% down.

Storage paths

To account for storage device redundancy, Event Management uses a separate impact calculation. You can see impact tree updates when the network storage topology changes from the business service. Event Management performs the following steps for alerts that contain storage CIs:

1. Create a business service map for the affected storage device:
   - Use the storage device in the sa_fs_to_storage_path table. The storage device definition uses the file system information in the path.
   - Use the elements in the storage path that derive from the Configuration Item (cmdb_ci) table. Also, use the elements that are associated to the path from the Infra Path To Elements (sa_infra_path_assoc) table.
   - Set the relationships. The application CI has a Depends on::Used by relationship on an element in the path that is defined in the CI Relationship (cmdb_rel_ci) table. In the relationship, the application CI is the parent and the element in the storage path is the child.

2. Calculate a separate severity for each regular element in the path. Each regular element in the path contributes its own severity to its ancestors up to the original application CI the path.

3. Use the redundancy rule to calculate redundant elements in the path by reducing the severity on the impacted CIs by one level. For example, if the severity is Critical, the redundancy rule decreases by one level to Major.

4. Create the impact tree. Mark the magnitude of an outage by 100% down, 60% affected, 40% impaired, or 20% impaired. If the items in two or more clusters are affected, the impact is 100% down.

Related CIs

As alerts generate for a CI, additional impact calculations run for related CIs. For example, additional impact calculations run for a business service dependency to a CI that is not actually
part of the business service. These related CIs are not discovered as part of the service. Instead, the related CIs are specified by an infrastructure relationship definition.

The following impact calculation flow operates for alerts with CIs that have a dependency to related CIs which are considered outside the business service. Event Management performs the following steps:

1. Derive relationships between the business service CIs and related CIs. Use the relationships, impact rules, and other data from the Infrastructure Relations (em_impact_infra_rel_def) table.
2. Add related CIs to the impact tree and alert list on the Event Management dashboard.
   - Use data from the Infrastructure Relationship (em_impact_infra_rel_def) table to show containment links to the host.
   - Use the Impact Status (em_impact_status) and Alert History (em_alert_history) tables to determine the status.

Impact rules

Impact rules, which are used for impact calculation, estimate the magnitude or severity of an outage based on affected CIs.

The Impact Rule (em_impact_rule) table contains impact rules that show the applicable CIs, business services, and settings for impact. The following default impact rules are available.

**Application Cluster Member**

Determines how application cluster members affect the overall impact of the cluster. For example, if a three-member cluster requires 70% Influence to set the severity for the entire cluster to **Major**, each member has 23% Influence (70% divided by 3). The severity of the entire cluster can only change to **Major** when all three members have a severity of **Major**.

**Inclusion**

Determines the impact on entities with a Contains relationship. This rule is read-only.

**Infrastructure Dependencies**

Determines the definition of impact propagation for CIs in infrastructure relationships.

**CI Business Service**

Determines how impact applies to parent or child entities that are part of a business service.

**CI Impact**

Applies to application services. Determines the relationship between service members. The impact from child to parent CIs is always 100%. For example, the parent impact severity is derived from the child CI with the highest severity.

**CI Parent in Application**

Sets impact only on the parent entity.

**Network Path**

Determines how impact applies to parent or child entities that are part of a traditional network.

**OS Cluster Member**

Determines how host cluster members affect the overall cluster status based on a percentage or number of cluster members. For example, if a three-host cluster requires 50% Influence to set the severity of **Major**, each member has 17% Influence (50% divided by 3). The severity of the entire

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
cluster can only change to **Major** when two or more cluster members have a severity of **Major**. The entire cluster is also considered to be down.

**Storage Path**

Determines how impact applies to parent or child entities that are part of a storage network.

**Properties**

In addition to configuring impact rules, you can configure properties for impact calculation. Configure these properties, as appropriate:

- Enable alert group support (`evt_mgmt.impact_calculation.alert_group_support`)
- Minimum time in seconds for checking CI maintenance: checks both the Status field on the CI and any change request schedule for the CI (`evt_mgmt.impact_maintenance.sleep_time_sec`)

**View the impact tree**

The impact tree shows the relationships between CIs and the relative percentage impact for each child CI. This information is available for both discovered services and application services.

Role required: `evt_mgmt_admin`, `evt_mgmt_operator`, or `evt_mgmt_user`

The impact tree shows the result of impact rules on CI parent and child relationships. The tree represents a business service map with CIs from the Service Configuration Item Associations (svc_ci_assoc) table. When related CIs are connected but not part of the business service map, the impact for the related CIs can also appear in the impact tree.

The numbers or percentages on a parent CI summarize overall severity based on alerts from child CIs. The parent severity is based on impact rule computations.

You can also learn about the impact tree from the following video tutorial.
1. Open the business service map for the service from either the Event Management dashboard or the Application service list.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>From the Event Management dashboard</td>
<td>1. Navigate to Event Management &gt; Dashboard.</td>
</tr>
<tr>
<td></td>
<td>2. Double-click the tile of the service.</td>
</tr>
<tr>
<td>From the Application service list</td>
<td>1. Navigate to Event Management &gt; Services &gt; Application Services.</td>
</tr>
<tr>
<td></td>
<td>2. Click View Map next to the service.</td>
</tr>
</tbody>
</table>

2. On the business service map, click a CI that has a severity color.
3. Click the Impact tab.
4. From the Impact Tree tab in the side pane, expand the parent CI and click the CI that generated alerts.
5. Use the severity color to locate CIs that contain alerts:
   - **Critical (red)**: Immediate action is required. The resource is either not functional or critical problems are imminent.
   - **Major (orange)**: Major functionality is severely impaired or performance has degraded.
   - **Minor (yellow)**: Partial, non-critical loss of functionality or performance degradation occurred.
   - **Warning (blue)**: Attention is required, even though the resource is still functional.
   - **Info (green)**: An alert is created. The resource is still functional.
6. Do one or more of the following:
### Adjust impact rules

- **Description:**
  1. Click the **Impact** tab.
  2. Review and adjust the impact rules as necessary.

### Remediate an alert

- **Description:**
  1. Click the **Alerts**.
  2. Right-click the alert and select **Run remediation**.

---

### Adjust impact rules for a CI

Configure impact rules to customize the impact calculation. The impact rules update the overall alert and show the impact on related CIs. When you change impact rules, the updates apply to alert severity in places such as the Event Management alerts console and dashboard.

**Role required:** evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

You can view and adjust the impact rules of CIs from the business service map of a service.

1. Open the business service map from either the Event Management dashboard or the Application service list.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>From the Event Management dashboard</strong></td>
<td></td>
</tr>
<tr>
<td>1. Navigate to <strong>Event Management</strong> &gt; <strong>Dashboard</strong>.</td>
<td></td>
</tr>
<tr>
<td>2. Double-click the tile of the application service.</td>
<td></td>
</tr>
<tr>
<td><strong>From the Application Service list</strong></td>
<td></td>
</tr>
<tr>
<td>1. Navigate to <strong>Event Management</strong> &gt; <strong>Services</strong> &gt; <strong>Application Services</strong>.</td>
<td></td>
</tr>
<tr>
<td>2. Click <strong>View Map</strong> next to the application service.</td>
<td></td>
</tr>
</tbody>
</table>

2. On the business service map, click a CI.

3. Below the map, click the **Impact** tab and then adjust the impact rules accordingly.
Impact Rules

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the impact rule.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Application Cluster Member</strong>: Determines how application cluster members affect the overall impact of the cluster. For example, if a three-member cluster requires <strong>70% Influence</strong> to set the severity for the entire cluster to <strong>Major</strong>, each member has <strong>23% Influence</strong> (70% divided by 3). The severity of the entire cluster can only change to <strong>Major</strong> when all three members have a severity of <strong>Major</strong>.</td>
</tr>
<tr>
<td></td>
<td>- <strong>CI Business Service</strong>: Determines how impact applies to parent or child entities that are part of a business service.</td>
</tr>
<tr>
<td></td>
<td>- <strong>CI Impact</strong>: Applies to application services. Determines the relationship between service members. The impact from child to parent CIs is always 100%. For example, the parent impact severity is derived from the child CI with the highest severity.</td>
</tr>
<tr>
<td></td>
<td>- <strong>CI Parent in Application</strong>: Sets impact only on the parent entity.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Inclusion</strong>: Determines the impact on entities with a Contains relationship. This rule is read-only.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Infrastructure Dependencies</strong>: Determines the definition of impact propagation for CIs in infrastructure relationships.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Network Path</strong>: Determines how impact applies to parent or child entities that are part of a traditional network.</td>
</tr>
<tr>
<td></td>
<td>- <strong>OS Cluster Member</strong>: Determines how host cluster members affect the overall cluster status based on a percentage or number of cluster members. For example, if a three-host cluster requires <strong>50% Influence</strong> to set the severity of <strong>Major</strong>, each member has <strong>17% Influence</strong> (50% divided by 3). The severity of the entire cluster can only change to <strong>Major</strong> when two or more cluster members have a severity of <strong>Major</strong>. The entire cluster is also considered to be down.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Storage Path</strong>: Determines how impact applies to parent or child entities that are part of a storage network.</td>
</tr>
<tr>
<td>Impact On</td>
<td>The valid type of service for this impact rule.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Business Service</strong>: The impact rule applies to a business service.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Parent</strong>: The impact rule applies to the parent CI.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Influence</td>
<td>The value to allow the impact rule to set on the parent. Set a value on the parent and each child. This field works with the Influence Units field.</td>
</tr>
<tr>
<td>Influence Units</td>
<td>The unit of measurement to show impact on the parent. Set the highest value on the parent CI. Set lower values on each child CI. When alerts occur for several child CIs, Event Management calculates the sum of the Influence Units of each affected child. If the sum exceeds the parent Influence Units value, the parent receives the highest severity in the set. This field works with the Influence field.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Percent</strong>: When the percentage exceeds the Influence value, apply the impact on the parent.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Number</strong>: When the number exceeds the Influence value, apply the impact on the parent.</td>
</tr>
<tr>
<td></td>
<td>For example, you can set the parent Influence Units to 100% and the child CIs to these values:</td>
</tr>
<tr>
<td></td>
<td>• Child 1: 40%</td>
</tr>
<tr>
<td></td>
<td>• Child 2: 40%</td>
</tr>
<tr>
<td></td>
<td>• Child 3: 40%</td>
</tr>
<tr>
<td></td>
<td>• Child 5: 70%</td>
</tr>
<tr>
<td></td>
<td>When alerts for Child 3 and Child 5 have the Critical severity, the severity of the parent is set to Critical because the sum is greater than 100% (40% + 70% = 110%).</td>
</tr>
<tr>
<td>Impact when Critical</td>
<td>The alternative impact to use when the calculated severity is Critical. If the column to the right of Impact when Critical has a higher severity, update all right-most columns with the same or a lower severity as Impact when Critical. To make sure the topology and impact tree accurately show impact, always update each column accordingly.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Critical</strong>: Red (highest severity).</td>
</tr>
<tr>
<td></td>
<td>• <strong>Major</strong>: Orange.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Minor</strong>: Yellow.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Warning</strong>: Blue (lowest severity).</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Impact when Major</td>
<td>The alternative impact to use when the calculated severity is <strong>Major</strong>. If the column to the right of Impact when Major has a higher severity, update all right-most columns the same or a lower severity as Impact when Major. To make sure the topology and impact tree accurately show impact, always update each column accordingly.</td>
</tr>
<tr>
<td></td>
<td>• Critical: Red (highest severity).</td>
</tr>
<tr>
<td></td>
<td>• Major: Orange.</td>
</tr>
<tr>
<td></td>
<td>• Minor: Yellow.</td>
</tr>
<tr>
<td></td>
<td>• Warning: Blue (lowest severity).</td>
</tr>
<tr>
<td>Impact when Minor</td>
<td>The alternative impact to use when the calculated severity is <strong>Minor</strong>. Controls the field values to the right of this field. If the Impact when Warning column has a higher severity, use the Impact when Minor value for both fields. To make sure the topology and impact tree accurately show impact, always update each column accordingly.</td>
</tr>
<tr>
<td></td>
<td>• Critical: Red (highest severity).</td>
</tr>
<tr>
<td></td>
<td>• Major: Orange.</td>
</tr>
<tr>
<td></td>
<td>• Minor: Yellow.</td>
</tr>
<tr>
<td></td>
<td>• Warning: Blue (lowest severity).</td>
</tr>
<tr>
<td>Impact when Warning</td>
<td>The alternative impact to use when the calculated severity is <strong>Warning</strong>. To make sure the topology and impact tree accurately show impact, make sure that the columns to the left have higher severities.</td>
</tr>
<tr>
<td></td>
<td>• Critical: Red (highest severity).</td>
</tr>
<tr>
<td></td>
<td>• Major: Orange.</td>
</tr>
<tr>
<td></td>
<td>• Minor: Yellow.</td>
</tr>
<tr>
<td></td>
<td>• Warning: Blue (lowest severity).</td>
</tr>
</tbody>
</table>

Review the changes on the impact tree. For example, if you changed a number or percentage influence for child CIs, review the impact tree updates accordingly.

**View alert impact on CIs in a service map**

The business service map shows active alerts for CIs and the relationships between CIs. The business service map is available for all application services.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

A business service map shows alerts with impacted CIs and CI interdependencies. For example, changes to a connection between a host and hypervisor appear on the business service map. As the business service map definition or the statuses of alerts change, the business service map, alert, and impact information updates accordingly. You can open a business service map from these places:

- From the Application services list, you can view business service maps for application services.
- From the Monitored services list, you can view business service maps for monitored services.
- From the Event Management dashboard, you can view business service maps for discovered services and application services.

The following icons are used in business service maps. The icon shapes are slightly different for application services.

**Business service map icons**

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="application_server_icon" /></td>
<td>The application server icon represents applications such as Microsoft IIS or SQL servers.</td>
</tr>
<tr>
<td><img src="image" alt="call_server_icon" /></td>
<td>The call server icon represents physical and VM computers and servers.</td>
</tr>
<tr>
<td><img src="image" alt="entry_point_icon" /></td>
<td>The entry point icon represents the network starting point. For example, Layer 3 devices appear toward the top of the map, and connected software and services appear near the end of the map.</td>
</tr>
<tr>
<td><img src="image" alt="redundancy_arrow_icon" /></td>
<td>The redundancy arrow shows the number of redundant CIs.</td>
</tr>
<tr>
<td><img src="image" alt="load_balancer_icon" /></td>
<td>The load balancer icon.</td>
</tr>
<tr>
<td><img src="image" alt="gray_connector_icon" /></td>
<td>The gray connector shows a relationship between CIs.</td>
</tr>
<tr>
<td><img src="image" alt="gray_box_icon" /></td>
<td>Each box represents a network CI. A gray box represents a CI with no active alerts. Information about the CI is hidden.</td>
</tr>
<tr>
<td><img src="image" alt="redundancy_icon" /></td>
<td>The redundancy icon hides multiple CIs that are designated as redundant.</td>
</tr>
</tbody>
</table>
An impacted CI displays the color that represents the severity of the alert associated with the CI.

- **Critical (red):** Immediate action is required. The resource is either not functional or critical problems are imminent.
- **Major (orange):** Major functionality is severely impaired or performance has degraded.
- **Minor (yellow):** Partial, non-critical loss of functionality or performance degradation occurred.
- **Warning (blue):** Attention is required, even though the resource is still functional.
- **Info (green):** An alert is created. The resource is still functional.
- **No color:** No active alerts.

The storage icon represents fiber channel, hard drives, or other data storage devices.

The web server icon represents related web services for the network such as NGINX or JBoss.

**Note:** Session timeout settings do not apply to this screen, so the session remains connected, even when there is no human interaction. If this setting is a concern, either log out or close the active tab in the browser.

1. Perform one of the following actions:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| To open a map of an application service or a business service | 1. Navigate to Event Management > Dashboard.  
2. Double-click the tile of the service. |

To open an application service map

1. Navigate to Event Management > Services > Application Services.  
2. Click View Map next to the service.  

Alternatively, if you are in the Map window in Service Mapping, click Monitor Service at the top of the window.

2. Do one or more of the following:

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
</table>
| View alerts for a CI by type and severity | In the business service map:  
1. Click a CI tile.  
2. Below the map, click the Alerts tab and review the listed alerts. |
<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>View changes to a CI in a discovered service</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Click a CI tile.</td>
</tr>
<tr>
<td></td>
<td>2. Below the map, click the Changes tab.</td>
</tr>
<tr>
<td>Show alert bindings to CIs</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Click a CI tile.</td>
</tr>
<tr>
<td></td>
<td>2. Click the menu icon, and turn on the Affected CI’s indicator.</td>
</tr>
<tr>
<td></td>
<td>3. Below the map, click the Affected CI’s tab.</td>
</tr>
<tr>
<td>View CI properties</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Click a CI tile.</td>
</tr>
<tr>
<td></td>
<td>2. In the side pane, click the Properties tab and review information about the CI.</td>
</tr>
<tr>
<td></td>
<td>3. If you want to view more detailed information, scroll to the end of the pane and click Detailed Properties.</td>
</tr>
<tr>
<td>Show impact rules for a CI</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Click a CI that has a severity.</td>
</tr>
<tr>
<td></td>
<td>2. Below the business service map, click the Impact tab and review the listed impact rules.</td>
</tr>
<tr>
<td>Show alert details for networks or storage for a business service</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Right-click a path between CIs.</td>
</tr>
<tr>
<td></td>
<td>2. Select Show network path or Show storage path.</td>
</tr>
<tr>
<td>Display additional information for CIs</td>
<td>In the business service map header:</td>
</tr>
<tr>
<td></td>
<td>1. Click the menu icon.</td>
</tr>
<tr>
<td></td>
<td>2. Turn on the Map Indicators for the additional information that you want to view.</td>
</tr>
<tr>
<td>Remediate a CI</td>
<td>In the topology, right-click the CI, and then select Remediation options.</td>
</tr>
</tbody>
</table>

Create an infrastructure relationship for related CIs
Infrastructures show CIs that are connected to a business service but are not necessary parts of the service. Infrastructure relationships are only available for business services.
Role required: evt_mgmt_admin

When you create a CI infrastructure relationship, the information is stored in the Infrastructure Relations (em_impact_infra_rel_def) table. When alerts generate, the related CI accompanies the business service information on the Event Management dashboard and in the impact tree. Additional information for the related CIs only appears on a related dependency view map for the business service. The following default infrastructure relationships are available.
### Default infrastructure relationships

<table>
<thead>
<tr>
<th>Infrastructure relationship</th>
<th>Impact rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_appl</td>
<td>OS Cluster Member</td>
<td>Shows alert impact between hardware and software applications.</td>
</tr>
<tr>
<td>cmdb_ci_esx_server</td>
<td>Infrastructure Dependencies</td>
<td>Shows alert impact between VCenter and ESX clusters.</td>
</tr>
<tr>
<td>cmdb_ci_kvm</td>
<td>Infrastructure Dependencies</td>
<td>Shows that alert impact on Linux Kernel-based Virtual Machine (KVM) connectivity.</td>
</tr>
<tr>
<td>cmdb_ci_vm_zones</td>
<td>Infrastructure Dependencies</td>
<td>Shows alert impact on the Solaris VM zones.</td>
</tr>
</tbody>
</table>

For example, based on the `cmdb_ci_vm_zones` infrastructure relationship definition, Event Management adds `ZoneServer@mmp1` to the business service. The Containment rule manages impact severity on alerts.
Related CIs appear on the BSM

1. Navigate to **Event Management > Settings > Infrastructure Relations**.
2. Click **New**.
3. Fill in the fields, as appropriate.

**Infrastructure Relations form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Child Type</td>
<td>The table that contains data about the child entity.</td>
</tr>
<tr>
<td>Parent Type</td>
<td>The table that contains data about the parent entity.</td>
</tr>
<tr>
<td>Relation Type</td>
<td>The relationship between the child and parent entities.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Impact Direction</td>
<td>The impacts direction to show on the business service map.</td>
</tr>
<tr>
<td></td>
<td>• <strong>From Child to Parent</strong>: When an alert is regarding a child, show the impact on the parent.</td>
</tr>
<tr>
<td></td>
<td>• <strong>From Parent to Child</strong>: When an alert is regarding a parent, show the impact on the child entity.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Impact Rule</td>
<td>The impact rule to calculate infrastructure relationships:</td>
</tr>
<tr>
<td></td>
<td>- <strong>OS Cluster Member</strong>: Determines how host cluster members affect the overall cluster status based on a percentage or number of cluster members. For example, if a three-host cluster requires <strong>50% Influence</strong> to set the severity of <strong>Major</strong>, each member has <strong>17% Influence</strong> (50% divided by 3). The severity of the entire cluster can only change to <strong>Major</strong> when two or more cluster members have a severity of <strong>Major</strong>. The entire cluster is also considered to be down.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Application Cluster Member</strong>: Determines how application cluster members affect the overall impact of the cluster. For example, if a three-member cluster requires <strong>70% Influence</strong> to set the severity for the entire cluster to <strong>Major</strong>, each member has <strong>23% Influence</strong> (70% divided by 3). The severity of the entire cluster can only change to <strong>Major</strong> when all three members have a severity of <strong>Major</strong>.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Infrastructure Dependencies</strong>: Determines the definition of impact propagation for CIs in infrastructure relationships.</td>
</tr>
<tr>
<td></td>
<td>- <strong>CI Business Service</strong>: Determines how impact applies to parent or child entities that are part of a business service.</td>
</tr>
<tr>
<td></td>
<td>- <strong>CI Parent in Application</strong>: Sets impact only on the parent entity.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Inclusion</strong>: Determines the impact on entities with a Contains relationship. This rule is read-only.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Network Path</strong>: Determines how impact applies to parent or child entities that are part of a traditional network.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Storage Path</strong>: Determines how impact applies to parent or child entities that are part of a storage network.</td>
</tr>
<tr>
<td></td>
<td>- <strong>CI Impact</strong>: Applies to application services. Determines the relationship between service members. The impact from child to parent CIs is always 100%. For example, the parent impact severity is derived from the child CI with the highest severity.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

*View impact of child service on parent service*

In the map view of a business service, the severity of a child service is propagated to its parent service.

Role required:
In the map view of an application or of a discovered service, you can see that the severity of a child service is propagated to the parent service.

A business service is affected by the severity of alerts that are directly associated with it and alerts on the CIs that are in the service. If the severity of a child service is higher than that of its parent service, on the map view the parent service shows the severity of the child service. In the map view, the **Impact Tree** list shows the CIs. The color of the CIs indicates the alert severity. The child service is also affected by business services that are under it.

![Example where the parent service shows the severity of the child service](image)

If a loop is created, for example, by connecting a parent service to a sub-service, this relationship can be viewed in the map view. The impact of this loop is not calculated and the Impact Tree does not display this relationship. For example, assume that there are three CIs, A, B, and C, where C is a sub-service to B and B is a sub-service to A. The relationship of the A connected service to C is displayed in the map view. However, this relationship is not shown in the Impact Tree.

**Note:** A message appears in the system error log in the following format, where the sys_id of the service and the node changes accordingly: Found loop in ServiceId:[sysId], and node:[sysId]. Removing this relation to avoid impact loops

For example, Found loop in ServiceId:409d4fd59f030200fe2ab0aec32e705c, and node:df5cf6159f030200fe2ab0aec32e70a2. Removing this relation to avoid impact loops

Click a service or CI to see the alerts that are associated with it. For example, if you click the high-level business service, the alerts that are associated with it are displayed in the alert area under the Map View. The alerts listed are those of the selected service. Alerts of child-services are listed when those services are selected.
Example of alerts listed under the Map View

In the Event Management dashboard, double-click the required application service or discovered service.

In the business service map, you can visually verify that the service health of the sub-service is rolled-up to the parent service. At the bottom of the icon of an impacted CI is a colored bar that represents the severity of the alert. This color bar also displays in the bottom bar of its parent service.

Alert group types

Alerts are correlated, or grouped, either automatically or manually into (R)ule-based, (A)utomated, (M)anual, or (C)MDB alert groups. Grouping alerts enables you to narrow down problems by focusing on the primary alerts in the correlated group.

You can learn about Event Management grouping of alerts from this video tutorial:

When evaluating incoming alerts to form alert groups, the types of group that alerts might potentially belong to, are considered in the following order. After an alert becomes part of a group, it is not available for any other group.

1. Rule-based
3. Automated
4. CMDB

You can view all alert groups by navigating to Event Management > Alert Console. The icon in the Group column denotes the alert group type. Alerts that do not have an entry in the Group column are not correlated with any group.

At any given time, an alert can only belong to a single alert group.

### Note:
The filter that defines Alert Groups should not be on fields that do not appear in the em_alert_history table because impact calculation is not calculated properly. This situation occurs because fields like Event Count, Priority, and Priority Group are not copied to the em_alert_history table for impact calculation.
### Alert group types

<table>
<thead>
<tr>
<th>Type</th>
<th>Icon</th>
<th>Description</th>
<th>More information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule-based</td>
<td>R</td>
<td>Related alerts that have been grouped according to compliance with alert correlation rules. Alert correlation rules are used to group alerts that are related.</td>
<td>Create an alert correlation rule</td>
</tr>
<tr>
<td>Automated</td>
<td>A</td>
<td>Automated groups are:</td>
<td>Automated alert groups</td>
</tr>
<tr>
<td></td>
<td></td>
<td>· Groups that are aggregated automatically by alert aggregation and RCA. A virtual alert is added to the group as the primary alert of the group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>· The Aggregated type of automated group is formed when two or more alerts have the same CI type and metric name.</td>
<td></td>
</tr>
<tr>
<td>Manual</td>
<td>M</td>
<td>Alerts that have been manually grouped.</td>
<td>Create a manual alert group</td>
</tr>
<tr>
<td>CMDB</td>
<td>C</td>
<td>Based on CI relationships in the CMDB, for CIs without historical data that could have been used to group alerts.</td>
<td>CMDB alert groups</td>
</tr>
</tbody>
</table>

Double-click the **Group** column for an alert group to open its **Grouped Alerts** dialog box, where you can:

- Display all alerts in the group.
- Provide feedback about the usefulness of the group.
- Manually add or remove alerts from the group.

**View timeline for alert groups**

A timeline format that displays alerts and alert groups that correspond to the alerts and alert groups currently visible in the list view in the alerts console.

A line of blue dots at the top of the timeline view represent all the alerts displayed in the list view, and all the alerts that are included in the alert groups displayed in the list view. Alert groups and alert bubbles are displayed underneath across a timeline, according to the update time of the alerts.

1. Navigate to **Event Management > Alert Console**.
2. Enable **Group Timeline** and do any of the following:
   - Double-click a group name in an alert group in the timeline view to display the set of alerts within the group. For details about the alert form, see [View alert information](#).
   - Hover over a blue dotted alert, or over an alerts bubble in a group to display further details.
   - Zoom into alert groups to change the time granularity of the timeline. Drag the mouse over a collection of alert groups or blue dotted alerts to highlight the area of the timeline to zoom into.
   - Use one of the time presets to display alerts from a certain period such as alerts from the past 7 days.
   - Click an alerts bubble to open the **Grouped Alerts** dialog box which displays the individual alerts in the group. This view is equivalent to clicking a group in the list view.
   - Page through the list view to display the timeline for another set of alerts and groups.
Add alert to an alert group
After viewing the individual alerts in an alert group such as an automated alert group, you might conclude that there is another alert that should be added to the group. You can manually add an alert to an existing alert group so the group is more complete and useful in troubleshooting an incident.

Role required: evt_mgmt_operator

If the alert that you choose to add to an alert group already belongs to an alert group, the alert is automatically removed from its current alert group, thereby changing the alert group to which the alert belongs.

The addition of an alert to an alert group is registered as a form of feedback to alert groups. Later, when similar alert groups are formed, the Service Analytics Learner automatically repeats these additions which were previously done manually by the user.

Note: A closed alert cannot be assigned as a parent.

1. Navigate to Event Management > Alert Console.
2. Locate the group to which you want to add alerts, and click the group parent alert to open its Alert form.
3. Underneath the Related Links section, select the Alerts tab.
4. Click Add To Group, and then select the alerts to add to the group.
   The Alerts (Add To Group view) view displays all alerts which can be added to the alert group. Primary and closed alerts, for example, are not displayed.
   Alternatively, you can add an alert to an alert group by changing the alert’s parent. When you specify a new parent for an alert, the alert is added to the new parent’s alert group, and gets removed from its current alert group. You can specify a new parent by modifying the Parent field on an Alert form, or by modifying the value in the Parent column of an alert in the Alert Console.
5. To remove an alert, click Actions on selected rows, and then click Remove.
   After viewing the individual alerts in an alert group, you may decide to remove an alert from the group to improve the group’s accuracy and usefulness in troubleshooting an incident.
   If an alert group is left with one or no alerts after removing alerts, the group is deleted and no longer displays in alert group lists. The virtual primary alert that was added when the group was created is closed.

On the Alert form, verify that the newly added alerts appear in the Alerts tab.
Provide feedback for an alert group
As you view the details of an automated or CMDB alert group in the Alert Console, you can provide feedback about the accuracy and usefulness of the group. This feedback can help future analysis, and over time, accumulated feedback continues to improve the accuracy of alert aggregation.

Role required: evt_mgmt_user

Provide feedback in the form of a Yes/No response to the question of how accurate and useful the alert group is. This form of feedback for an alert group is provided for the primary alert of the group, as described in the following procedure.

Another form of feedback is user manipulation of manual alert groups and automated alert groups:

- The Alert Aggregation Learner tracks manual additions and removals of alerts from automated alert groups. In subsequent occurrences of the same types of alerts with similar time correlation - this form of feedback is applied. Alert aggregation and RCA automatically repeats the
addition or removal actions that were previously done manually. If you undo any previous alert additions or removals to those alert groups, the automatic process is adjusted accordingly.

You can review user additions and removals of alerts from an automated alert group, and undo any action so it is not automatically repeated by alert aggregation and RCA. See Edit feedback for an automated alert group for more information.

- The Alert Aggregation Learner also learns the patterns of alerts in manual alert groups. Later, when new streams of alerts arrive, alert aggregation and RCA automatically forms automatic alert groups according on these patterns.

To enable the application of this form of feedback during alert aggregation, set the property sa_analytics.agg.query_customer_feedback_enabled to true.

1. Navigate to Event Management > Alert Console.
2. Locate the group for which you want to provide feedback, and click its button to display the alerts included in the group.
3. Click the primary alert of the group to open its Alert form.
4. Based on your analysis of the accuracy and helpfulness of this group, set Feedback to Yes or No.
5. Click Update.

Edit feedback for an automated alert group
Modify which manual additions and removals of alerts are automatically applied to newly formed automated alert groups.

Event Management alert aggregation and RCA tracks your feedback of manually adding or removing alerts from automated alert groups. Later, when alert aggregation and RCA forms new automated alert groups, those additions and removals are automatically repeated.

You can edit this form of feedback by reviewing the list of user actions for the alert group, and then undoing specific alert additions and alert removals. Those additions and removals are not repeated when new automated alert groups are formed.

1. Navigate to Event Management > Alert Aggregation and RCA > Edit Feedback.
2. In the Grouped Alerts page:
   a) Click the alert group for which you want to edit feedback, and then click the user edit (Added or Removed) that you want to undo.
   b) Or, click the button next to the alert group for which you want to edit feedback. In the list of all user edits (Added and Removed), click the user edit that you want to undo.
3. Fill out the form and then click Submit.
   Most fields on the form are automatically populated according to the selected alert group and the user edit.

Grouped Alert form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert</td>
<td>Alert that was manually added and that should not be automatically added in newly formed automated alert groups. This field is automatically set.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Aggregated Group</td>
<td>Automated alert group to which <strong>Alert</strong> was added or removed from. This field is automatically set.</td>
</tr>
<tr>
<td>User Edits</td>
<td>User action of <strong>Added</strong> or <strong>Removed</strong> for <strong>Alert</strong>. This field is automatically set.</td>
</tr>
<tr>
<td>Undo User Edits</td>
<td>Check box for preventing the automatic addition or removal of alerts that are similar to <strong>Alert</strong>, when forming alert groups that are similar to <strong>Aggregated Group</strong>.</td>
</tr>
<tr>
<td>Primary Alert</td>
<td>Primary alert of <strong>Aggregated Group</strong>. This field is automatically set.</td>
</tr>
<tr>
<td>Group</td>
<td>Type of the alert group, such as CMDB, manual, or automated.</td>
</tr>
<tr>
<td>Last event generation time</td>
<td>The last time the event that is linked to the alert occurred. This time is the ServiceNow processing time, not the source system time. This field is automatically set.</td>
</tr>
<tr>
<td>Metric Name</td>
<td>Unique name that describes which metric data is collected. This field is automatically set.</td>
</tr>
<tr>
<td>Resource</td>
<td>Node resource that is relevant to the event that is linked to the alert. This field is automatically set.</td>
</tr>
<tr>
<td>Severity</td>
<td>The severity of the event. The value for this field is copied from the event unless the event closes the alert, in which case the previous severity is retained for reporting.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Critical</strong>: Immediate action is required. The resource is either not functional or critical problems are imminent.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Major</strong>: Major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Minor</strong>: Partial, non-critical loss of functionality or performance degradation occurred.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Warning</strong>: Attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Info</strong>: An alert is created. The resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Clear</strong>: No action is required. An alert is not created from this event. Existing alerts are closed.</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits. This field is automatically set.</td>
</tr>
</tbody>
</table>
Create a manual alert group
After you manually create a manual alert group, you can continue and manually add specific alerts to the group. The resultant manual alert group can contain many alerts and is useful in troubleshooting an incident.

Role required: evt_mgmt_operator

To initially create a manual alert group, select at least two alerts that do not belong to any alert group. Make one of those alerts the parent of the other alert. The new manual alert group appears in the Alert Console.

After manually creating a manual alert group, the Alert Aggregation Learner learns the group and generates a pattern based on the alerts in the group. Later, if alert aggregation and RCA detects a similar pattern in incoming streams of alerts, then alert aggregation and RCA forms automatic alert group according to the pattern.

Note: At any given time, an alert can belong only to a single alert group. A closed alert cannot be assigned as a parent.

1. Navigate to Event Management > Alert Console.
2. Clear the Correlated Alerts toggle. All alerts appear in the list, including open, primary, and secondary alerts.
3. Note the alert number of an alert that you want to set as the parent in a new alert. Ensure that the alert does not belong to any alert group. This alert is also known as the primary alert.
4. Click an alert that you want to set as the secondary alert in the new manual alert group. Ensure that the alert does not belong to any alert group.
5. In the Parent field of the selected alert, enter the alert number of the alert that is being configured as the primary alert.
6. Click Submit.

When the Alert Consolerefreshes, the primary alert that was selected, is denoted as being in the manual alert group. The alert, that is a member of the new manual alert group, is denoted as being the secondary alert.

In the Alert Console, you can add and remove other alerts to the manual alert group.

View reasons for alert groupings
As alerts are added to a group, a message is added to the alert’s Work notes field that indicates the reason for aggregating that alert into the group.

Ensure that the evt_mgmt.alert_groups_reasoning.enable_worknotes property is configured with Value = all. Configure this property by navigating to System Properties > All Properties. The default values in this property are configured in the Choices field and are listed in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Alerts</td>
<td>Work note information is generated for all alerts.</td>
</tr>
<tr>
<td>Primary Alerts</td>
<td>Work note information is generated only for primary alerts.</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>Secondary Alerts</td>
<td>Work note information is generated only for secondary alerts.</td>
</tr>
<tr>
<td>None</td>
<td>No work note information is generated.</td>
</tr>
</tbody>
</table>

By default Value = all, indicating that work note information is generated for all alerts.

Role required: evt_mgmt_admin

In an alert record, the Work notes field in the Activities tab provides information that enables you to better understand the group to which the alert belongs. You can view the list of alerts in:

- Event Management dashboard
- Alert Intelligence
- Alert Console
- All Alerts

When another alert is added to an alert group, this information is shown in the Work notes field of the selected alert. When an alert is removed from the group, this information is also shown in the Work notes field of the selected alert.

Information related to the following types of alert groups appears in Work notes fields:

- Secondary
- Automated
- Aggregated
- CMDB

1. Navigate to Event Management > All Alerts.
2. Click an alert record that is part of a group.
3. Click the Activities tab.
   In the Work notes field, you can see the reason for aggregating an alert into a group.
   The following example provides a sample explanatory note.

   2018-10-16 14:53:14 - System Administrator
   Alerts Alert0056912, Alert0056913 were grouped together due to pattern

   To see the patterns that caused the alerts to be grouped, you would click the pattern hyperlink.

Alert monitoring

Alert generation and remediation can be monitored from the Event Management dashboards, alert console, and alert timeline.

- The Alerts Console shows incoming alerts for discovered business services, application services, technical services, and alert groups.
- The Event Management Dashboard displays the health of services. It consolidates alert information about discovered business services, application services, technical services, automated alert groups, and other alert groups. Additional information is available on the business service map, impact tree, and timeline.
• The **Overview** dashboard displays reports on active alerts. You can drill down to individual alerts from these reports.

**Monitor service health**

On the Event Management dashboard, you can view alerts by business service, application service, technical service, and alert group. For services, you can also open a business service map to view relationships between CIs in the service.

Ask an administrator to create at least one business service, application service, technical service, or alert group if none of these items exist.

**Role required:** evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

The dashboard tiles represent a service, alert group, or CI. Tile sizes correspond to the selection in the **Prioritize by** field in the dashboard. A larger tile size represents the severity of the alerts. The tile color represents the overall alert severity.

You can learn about the Event Management basic components from this video tutorial:

**These tile icons are available:**

<table>
<thead>
<tr>
<th>Dashboard tile icons</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Icon" /></td>
</tr>
<tr>
<td><img src="image" alt="Icon" /></td>
</tr>
<tr>
<td><img src="image" alt="Icon" /></td>
</tr>
<tr>
<td><img src="image" alt="Icon" /></td>
</tr>
</tbody>
</table>

Each tile represents the highest severity of an alert for the service, alert group, or CI. For example, a tile appears in red if an alert is critical. A green tile indicates either information alerts or no currently active alerts. As the alert severities change, the dashboard updates accordingly. The following severities are available:

- **Red:** critical severity (highest severity). Immediate action is required. The resource is either not functional or critical problems are imminent.
- **Orange:** major severity. Major functionality is severely impaired or performance has degraded.
- **Yellow:** minor severity. Partial, non-critical loss of functionality or performance degradation occurred.
- **Blue:** warning severity (lowest severity). Attention is required, even though the resource is still functional.
- **Green:** informational. No severity. An alert is created. The resource is still functional.

To integrate with Operational Intelligence, activate the Operational Intelligence (com.snc.sa.metric) plugin and follow the steps in the [Get started with Operational Intelligence](#) topic. In the dashboard, double-click a discovered business service to drill down and reach the **topology map**. These Operational Intelligence buttons are available:
• **Click View metrics.** This view displays a canvas to which you can drag CIs and display metric charts on. The server CIs are listed in the device pane. Point to the CIs in the list to display a tooltip that notes if the CIs have no metric data. For CIs that do have metric data, you can drill down into that data. Use the View metrics button to switch between the map and metric chart viewer. You can drag a CI that has metric data into the canvas area in the map to create a chart. The metric data also appears on the timeline.

• The **Metrics** button is automatically selected when **View metrics** is clicked. In the device pane, select a server. All the CIs that are related to that server are listed and metric data is displayed for those CIs that have metric data. A circle that depicts the metric anomaly score appears to the right of each metric value, see Configure anomaly score thresholds. If it is within the green band range, Operational Intelligence does not track changes to an anomaly score. The colors of the dots at the upper left on the CI icons or the absence of a dot, signify:

  • Grey dot: Metrics are being tracked for the CI, but none of those metrics have ever had an anomaly.
  • Colored dot: Corresponds to the last recorded anomaly score according to the configuration of the color band ranges.
  • No dot: There are no metrics for the CI and there is no data to drill into.

• Items can be selected on the map to filter the list of CIs on the right.

• The time range for the graphs is tied to the timeline at the top of the service map.

For more information, see View metric values in the Insights Explorer.

**Note:** Session timeout settings do not apply to this screen, so the session remains connected, even when there is no human interaction. If this setting is a concern, either log out or close the active tab in the browser.

1. Navigate to **Event Management > Dashboard.**
2. Choose from:
   - To view alerts by business service, application service, technical service, or alert group, click **Services**.
• To view alerts in your business group only, click **Groups**.

3. **Do one or more of these actions:**

<table>
<thead>
<tr>
<th>Task</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prioritize alerts</strong></td>
<td>Above the tile area:</td>
</tr>
<tr>
<td></td>
<td>1. Click <strong>Prioritize by list</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. Select <strong>Severity</strong>, <strong>Business criticality</strong>, or <strong>Cost</strong>.</td>
</tr>
<tr>
<td></td>
<td>The filter values use the Dashboard filter dictionary attribute from the <code>cmdb_ci_service_auto.list</code> to prioritize alerts.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>View alert details for a service or group</strong></th>
<th>In the tile area:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1. Click the service or group tile.</td>
</tr>
<tr>
<td></td>
<td>2. Review the list of corresponding alerts under the <strong>Alerts</strong> tab.</td>
</tr>
<tr>
<td></td>
<td>3. To view the properties of an alert, double-click it.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Show or hide alerts by severity</strong></th>
<th>In the dashboard header, click the icon of the lowest severity that you want to view in the dashboard.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>You can also move the slider to the left of the icon. The dashboard appears only the severities that are to the right of the slider. For example, move the slider to the left of the red icon if you want to view only alerts with critical severity.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>View the properties of an alert</strong></th>
<th>Click <strong>Alerts</strong> or <strong>Correlated Alerts</strong>, then double-click the required alert number.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Note:</strong> Alerts of type A (analytic alert group) do not show in the bottom alerts panel when <strong>Correlated Alerts</strong> is not selected.</td>
<td></td>
</tr>
</tbody>
</table>

| **View alerts from another service or group** | In the dashboard header, click the down arrow, and then select the service or group name. |

| **Show active alerts for a group**          | In the dashboard header, click **Service**. Then click the group name and review the alerts listed for it under the **Alerts** tab. |

| **Show alert CI relationships or bindings in a business service map** | On the dashboard, double-click a tile for the service. For more information on the business service map, click **View alert impact on CIs in a service map**. |

<p>| <strong>Show an aggregation of correlated alerts for services and alert groups</strong> | Below the dashboard tiles, click <strong>Correlated Alert</strong>. |</p>
<table>
<thead>
<tr>
<th>Task</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remediate an alert</td>
<td>Under the <strong>Alerts</strong> tab, right-click the alert and then select <strong>Run remediation</strong>.</td>
</tr>
</tbody>
</table>

4. To reduce the number of alerts that display in the alerts panel, enable **Correlated Alerts**.

The alerts panel displays alert groups and alerts that have not been grouped.

![Correlated Alerts](image)

**Note:** The **Correlated Alerts** toggle also appears in the **Alerts Console**. When either the **Alerts Console** or the dashboard is opened, the **Correlated Alerts** toggle opens in the setting that was last selected.

---

**Create a customized dashboard view**

You can create a customized Event Management dashboard view and save it as a favorite link in the navigation tree.

**Role required:** evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

You can filter business services and save the customized dashboard view as a favorite link.

**Note:**
- The Dashboard View feature applies only to business services, and not to business service groups. The business service groups appear whether they do or do not match the filter, and whether they do or do not contain any services that match the filter.
- The Dashboard View filters services in the dashboard. However, alerts are not filtered. Alerts appear whether they do or do not match the filter.

1. Navigate to **Event Management > Settings > Dashboard Views**.
2. In the **Dashboard Views** screen, click **New**.
3. In the **Name** field, enter a descriptive name for the dashboard view.
4. Select **Active**.
5. Specify the filter conditions and click **Submit**.
6. Click the options menu in the title next to **Dashboard Views**.
7. In the context menu, click **Create Favorite**.

In the **Create Favorite** screen, the name that you specified for the view is displayed. You can optionally edit the name. Select a symbol and its color to display to the left of the dashboard view name.
8. Click **Done**.
   The dashboard view name and its symbol appear in the navigation tree.

9. In the navigation tree, click the required dashboard view name to show the customized dashboard.
   You can view a selected service group in the Event Management dashboard by pasting a link into a browser, using this format: `https://<instance_name>.service-now.com/nav_to.do?uri=%2F$sw_dashboards.do?sysparm_cancelable=true&amp;groupId=<GROUP_SYSID>&tree_state=open&amp;sysparm_bsid=<GROUP_SYSID>&openDirectGroup=true`.
   Replace the variables `<instance_name>` and `<GROUP_SYSID>` with the respective values of the selected service group. Copy and then paste the link into a browser to view the selected service group in an Event Management dashboard.

   For example, if the instance is named `asteroid` and the sys_id of the service group is `17f8dc780fd22300b678778ce1050eb4`, copy and paste this link into a browser:
   ```
   http://asteroid.service-now.com/nav_to.do?uri=%2F$sw_dashboards.do?sysparm_cancelable=true&amp;groupId=17f8dc780fd22300b678778ce1050eb4&amp;tree_state=open&amp;sysparm_bsid=17f8dc780fd22300b678778ce1050eb4&amp;openDirectGroup=true
   ```

**View monitored services**

View all services that Event Management supports, such as, alert groups, discovered services, application services, and technical services. According to the type of service, you can view service definition details or drill down into the service.

Role required: `evt_mgmt_user`

1. Navigate to **Event Management > Monitored services**.
2. Click **View Service** to open the view that is relevant to that service.
   When you drill down to the contents of the service, either a map, alert list, or CI list appears according to the type of service.
   For example, if you click **View Service**, for SNMP1000 you can drill down to the contents of the service and the related topology map appears.
3. To view service definition details of a service, click the name of the required service.

For example, in the Name column, click SNMP1000.

**Monitor and manage alerts**

Monitor and manage alerts in Alert Intelligence in Workspace. You can also monitor and manage alerts in the Alerts Console.
You can monitor and manage alerts in the following components:

- The alerts list in Alert Intelligence and in the Alerts Console shows incoming alerts for discovered business services, application services, technical services, and alert groups.
- The Event Management Dashboard displays the health of services. It consolidates alert information about discovered business services, application services, technical services, automated alert groups, and other alert groups. Additional information is available on the business service map, impact tree, and timeline.
- The Overview dashboard displays reports on active alerts. You can drill down to individual alerts from these reports.

Monitor incoming alerts

You can monitor incoming alerts in Alert Intelligence in Workspace. You can also monitor alerts in the Alerts Console.

You can learn about Event Management basics, including the Alerts Console, from the following video.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

1. Navigate to Event Management > Alert Intelligence.
2. You can also navigate to Event Management > Alert Console.

<table>
<thead>
<tr>
<th>Column heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>Unique ID generated by Event Management to identify the alert.</td>
</tr>
<tr>
<td>Column heading</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Group          | An entry in this column indicates that the associated alert is a member of an alert group. Alerts that do not have an entry in this column are ungrouped alerts.  
- **CMDB**: CIs without historical data that were aggregated by alert aggregation and RCA based on CI relationships in the CMDB.  
- **Manual**: This alert is a member of an alert group that is formed when right-clicking an alert and setting it as secondary to the selected primary alert.  
- **Secondary**: This alert is a component of an alert group. The alert at the head of the group is known as the primary alert. When **Correlated Alerts** is selected, the secondary alerts that are under the primary alert do not display, making the Alerts Console less cluttered and easier to review.  
- **Blank**: This alert is an ungrouped alert. To make an ungrouped alert become a member of a group, right-click it and select in the topic **Add to Groups**. Select the alert and click **Add Selected**.  
- **Automated**: Aggregated automatically by alert aggregation and RCA. A virtual alert is added to the group as the primary alert of the group.  
- **Rule**: Alert group created as a result of a user configured correlation rule. |
| Severity        | The severity of the event. The value for this field is copied from the event unless the event closes the alert, in which case the previous severity is retained for reporting.  
- **Critical**: Immediate action is required. The resource is either not functional or critical problems are imminent.  
- **Major**: Major functionality is severely impaired or performance has degraded.  
- **Minor**: Partial, non-critical loss of functionality or performance degradation occurred.  
- **Warning**: Attention is required, even though the resource is still functional.  
- **Info**: An alert is created. The resource is still functional.  
- **Clear**: No action is required. An alert is not created from this event. Existing alerts are closed. |
<p>| Priority group  | Indicates which alerts should be attended to first. Priority is calculated for each open alert and then mapped into one of four priority categories. |</p>
<table>
<thead>
<tr>
<th>Column heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>Value providing a guide as to the priority of the alert, based on a number of accumulated categories. For example, alert state and business criticality.</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, for example, SolarWinds or SCOM. Optionally, you can enter a description, for example, Group Alert. This field has a maximum length of 100.</td>
</tr>
<tr>
<td>Description</td>
<td>The alert description.</td>
</tr>
<tr>
<td>Node</td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>JSON string that represents a configuration item. For example, {&quot;name&quot;:&quot;SAP ORA01&quot;,&quot;type&quot;:&quot;Oracle&quot;}. The CI identifier that generated the event appears in the Additional information field. This field has a maximum length of 1000. Click ![icon] to open the alert in dependency view.</td>
</tr>
<tr>
<td>Impacted Services</td>
<td>Indicates the number of business services affected by this alert group. For example, an alert with a severity status of <strong>Major</strong>, might affect eight business services. Whereas, an alert with a severity status of <strong>Critical</strong>, might affect one business service.</td>
</tr>
<tr>
<td>Metric Name</td>
<td>Unique name that describes which metrics are collected and for which this alert has been created.</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Shows whether the resource affected by the alert is in maintenance. Valid values are <strong>true</strong> or <strong>false</strong>.</td>
</tr>
<tr>
<td>Task</td>
<td>The corresponding task for the alert, such as an incident, change, or problem.</td>
</tr>
<tr>
<td>Parent</td>
<td>Reference to a parent alert.</td>
</tr>
<tr>
<td>Initial event generation time</td>
<td>Time that the initial event occurred in the remote system.</td>
</tr>
</tbody>
</table>

If Operational Intelligence is activated, you can right-click an alert and click **View Metrics** to open the integrated Insights Explorer and Dependency Views map for the CI that is associated with the alert.

Manage alerts
Alerts can be managed in Alert Intelligence in Workspace. You can also manage alerts in the Alerts Console.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user
1. Navigate to Event Management > Alert Intelligence

For information about how to manage alerts in Workspace, see Manage alert lists in Workspace.

2. To manage alerts, you can also navigate to Event Management > Alert Console.

3. In Alert Console list view, right-click the required alert record and select any of the following options.

<table>
<thead>
<tr>
<th>Alert right-click options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy URL to Clipboard</td>
<td>The URL is copied to your clipboard. For more information about the URL, see Launch web application from alert. On some browsers, you are prompted to manually copy the URL to the clipboard. The Now Platform frame around the dashboard is not included in the link.</td>
</tr>
<tr>
<td>Copy sys_id</td>
<td>The SYSId of the alert is copied to the clipboard.</td>
</tr>
<tr>
<td>Assign Tag</td>
<td>Create and assign tags to specific records to group and organize alerts.</td>
</tr>
<tr>
<td>Quick Response</td>
<td>Either apply remediation to the alert or launch a web application.</td>
</tr>
<tr>
<td>Acknowledge</td>
<td>Selects the Acknowledge check box on the Alert form to indicate that the alert is acknowledged.</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Selects the Maintenance check box on the Alert form to indicate that the related CI is receiving maintenance. Selecting this check box does not change the Status field of the CI.</td>
</tr>
<tr>
<td>Close</td>
<td>Marks the alert as closed to indicate that the alert is resolved and requires no further action.</td>
</tr>
<tr>
<td>View Metrics</td>
<td>Opens a dependency views map in metrics mode for the CI that is associated with the alert. The metrics data of the CI is displayed, and you can use the Insights Explorer to build metric charts.</td>
</tr>
</tbody>
</table>

Note: You cannot report alerts directly on business services groups.
4. To view the alerts in a group, click the play icon next to a parent alert.

Monitor alerts for an application service

To view information for application services only, navigate to the Application service list. From this list, you can open business service maps to view and manage alerts for the CIs in each service.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

1. Navigate to Event Management > Services > Application Services.

2. In the row of the required application service, click View Map.

3. Do one or more of these actions:

<table>
<thead>
<tr>
<th>Option</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>View alert details for a CI</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Click a CI tile.</td>
</tr>
<tr>
<td></td>
<td>2. Below the map, click the Alerts tab and review the listed alerts.</td>
</tr>
<tr>
<td>View impact on the CI parent</td>
<td>In the business service map:</td>
</tr>
<tr>
<td></td>
<td>1. Click a CI tile.</td>
</tr>
<tr>
<td></td>
<td>2. Below the business service map, click the Impact tab and review the listed impact rules.</td>
</tr>
<tr>
<td></td>
<td>3. Adjust the impact rules as necessary.</td>
</tr>
<tr>
<td>Change the map display, map layout, or map indicators</td>
<td>In the business service map header:</td>
</tr>
<tr>
<td></td>
<td>1. Click the menu icon.</td>
</tr>
<tr>
<td></td>
<td>2. Configure the appropriate settings.</td>
</tr>
<tr>
<td>Option</td>
<td>Action</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>Navigate to another application service</td>
<td>In the business service map header: 1. Click the down arrow next to the service name or the folder icon for all services. 2. Search for and select another application service ( ). The icon color represents the highest impact or severity for active alerts on the application service. - <strong>Critical</strong>: Red (highest severity). - <strong>Major</strong>: Orange. - <strong>Minor</strong>: Yellow. - <strong>Warning</strong>: Blue (lowest severity).</td>
</tr>
</tbody>
</table>

**View properties for a CI**

<table>
<thead>
<tr>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>In the business service map: 1. Click a CI tile. 2. In the side pane, click the <strong>Properties</strong> tab and review information about the CI. 3. If you want to view more detailed information, scroll to the end of the pane and click <strong>Detailed Properties</strong>.</td>
</tr>
</tbody>
</table>

---

**Priority group**

For better triage and focus, alerts that have a higher priority are brought to the top of the alert list. This placement brings to your attention those alerts that require you to handle them at a higher priority than other alerts.

The priority group indicates which alerts should be attended to first. Priority is calculated for each open alert and then mapped into one of four priority categories.

**Thresholds**

Thresholds on the calculated priority are used to determine to which category the alert is mapped to. Alerts with a priority above the value of the `evt_mgmt.top_priority_group_threshold` property are categorized as **Urgent**, which is the highest priority. The default value of this threshold is 1M.

**Priority group categories**

The priority groups into which alerts fall are listed in the following table.

<table>
<thead>
<tr>
<th>Alert priority groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority group</td>
</tr>
<tr>
<td>Urgent</td>
</tr>
<tr>
<td>High</td>
</tr>
</tbody>
</table>
The value of the alert priority group is more meaningful than the alert severity on its own. For example, an alert with a severity of Major, may not be in need of resolution ahead of an alert whose priority is Critical. The combined group priority value takes the weight of each factor into account, giving you a better indication of which alert to attend to first.

The alert priority group is available in alert lists in the Dashboard, Alert Console, and All Alerts page. For example, in the All Alerts list, the alert priority group is displayed as depicted in the following image.

<table>
<thead>
<tr>
<th>Number</th>
<th>Group</th>
<th>Severity</th>
<th>Priority group</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demo0001011</td>
<td></td>
<td>Warning</td>
<td>Low</td>
<td>100</td>
</tr>
<tr>
<td>Demo0001006</td>
<td></td>
<td>Warning</td>
<td>Moderate</td>
<td>100</td>
</tr>
<tr>
<td>Alert0010004</td>
<td></td>
<td>Minor</td>
<td>High</td>
<td>200</td>
</tr>
</tbody>
</table>

**Triggers that cause recalculation**

Changes to open alerts trigger the recalculation of the priority and then the alert is again categorized into the relevant priority group. Closed alerts are not considered for alert priority group calculation.

**Alert priority**

Determine the order in which to handle alerts according to the alert priority score. Multiple factors determine the alert priority score and this value changes with changes to the underlying factors.

**Multi-factor decisions**

The value of the alert priority score is a composite of the value of the category and its relative weight. You can configure the factors that make up the score, as described in the following sections, according to Category order, Category value mapping, and CI type weight factor mapping. The categories are:

<table>
<thead>
<tr>
<th>Priority group</th>
<th>Icon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moderate</td>
<td><img src="image" alt="Moderate Icon" /></td>
</tr>
<tr>
<td>Low</td>
<td><img src="image" alt="Low Icon" /></td>
</tr>
</tbody>
</table>
- **Accumulated categories** - These categories accumulate items with different sub-types that may have a different status or weighing factor. For example, business services that are factored according to different business criticality or business services that are factored according to cost.

- **Choice-list categories** - A list of items. You can map any of the items to a different value, thereby changing the priority for that item. For example, critical severity, which has the value of 1 by default, can be mapped to 4 to give it a higher weight in the calculated alert priority score.

### List of categories

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Accumulated categories</strong></td>
<td></td>
</tr>
<tr>
<td>Number of impacted business-services</td>
<td>This category is factored by the business-criticality of the service. An alert that has an impact on one service has a lower priority score than an alert that has an impact on many services. For example, if a router is down, then it might affect many services and a higher priority ensures that this alert is handled before handling another alert that has an impact on fewer services. An alert that has an impact on many non-critical business services must be handled with priority. However, if an alert has an impact on only a few services, but they are critical services, then this alert has a higher priority.</td>
</tr>
<tr>
<td>CI type</td>
<td>This category distinguishes between various CI types. For example, an alert on a CI type that is a router or switch may have a higher priority than an alert on another CI type, like a server or application.</td>
</tr>
<tr>
<td>Number of Secondary alerts</td>
<td>Alerts that have characteristics similar to other alerts are usually grouped, for example, alerts with the same message key are grouped. A parent alert which has many secondary alerts has a higher priority than a parent alert that has only a few secondary alerts. All secondary alerts of a parent alert are counted. If a secondary alert is closed, it is no longer correlated to the parent alert. If a parent alert is closed, all secondary alerts are no longer correlated and their priority value is calculated accordingly.</td>
</tr>
<tr>
<td><strong>Choice-list categories</strong></td>
<td></td>
</tr>
<tr>
<td>Alert state</td>
<td>Priority score is not calculated for an alert that is in <strong>Closed</strong> state. The available states with their default values are:</td>
</tr>
<tr>
<td></td>
<td>- Open = mapping value 1</td>
</tr>
<tr>
<td></td>
<td>- Reopen = mapping value 2</td>
</tr>
<tr>
<td></td>
<td>- Flapping = mapping value 3</td>
</tr>
<tr>
<td>Alert severity</td>
<td>The priority score reflects the level of seriousness of the underlying alert, for example, an alert that has severity of <strong>Critical</strong> has a higher weighted value than an alert with a severity of <strong>Minor</strong>. Priority is not calculated for an alert that has a severity of <strong>Info</strong>. The available levels are:</td>
</tr>
<tr>
<td></td>
<td>- Critical</td>
</tr>
<tr>
<td></td>
<td>- Major</td>
</tr>
<tr>
<td></td>
<td>- Minor</td>
</tr>
<tr>
<td></td>
<td>- Warning</td>
</tr>
<tr>
<td>Category</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Role     | The priority score reflects the role of the alert. The priority score is higher for an alert that has a role of Parent, next is the role of None, and the least weight is given to an alert with a role of Secondary. A parent alert that has many secondary alerts has a higher value than a parent alert that has less secondary alerts. The available roles are:  
|          | None  |
|          | Parent|
|          | Secondary |

**Structure of the alert priority score**

**Weighted value for each category**

Each category has its own placement in the alert priority score, according to its order and limit. You can configure the order and limit, see Modify the alert priority score. There are only positive numbers.

**Example constituents of alert priority score of 24302020.003**

<table>
<thead>
<tr>
<th>Category</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business services</td>
<td>(24.0, 1000000.0)</td>
</tr>
<tr>
<td></td>
<td>For example, if there are 24 impacted business services, the computed priority value is 24,000,000 (24 * 1000000.0)</td>
</tr>
<tr>
<td>Severity</td>
<td>(3.0, 100000.0)</td>
</tr>
<tr>
<td>CI type</td>
<td>(20.0, 100.0)</td>
</tr>
<tr>
<td>Role</td>
<td>(2.0, 10.0)</td>
</tr>
<tr>
<td>Secondary</td>
<td>(0)</td>
</tr>
<tr>
<td>State</td>
<td>(3.0, 0.001)</td>
</tr>
</tbody>
</table>

Using the scores from the preceding table, the alert priority score is computed as follows:

$$ (24 \times 1000000) + (3 \times 10000) + (20 \times 100) + (2 \times 10) + (0) + (3 \times 0.001) = 24302020.003 $$

Replacing the values of the categories from the preceding table into the formula:

$$ (24 \times 1000000) + (3 \times 10000) + (20 \times 100) + (2 \times 10) + (0) + (3 \times 0.001) = 24302020.003 $$

In the Priority Breakdown area of the Additional Information tab, the alert priority value is displayed as 24302020.003. In the Alerts list, this value is displayed as 24302.

**Effect of category limits**

Limits placed on categories enable overflow values to affect the next category that is ranked higher in priority order. Each accumulated category has a predefined limit. If the count goes above that limit, 1 is added to the next higher-order category. For example, if the number of CI types goes over the limit, then 1 is added to the next category (Severity) which is higher in priority: \( yyy - zzz \) becomes \( yy(y+1) - 000 \)
Display in All Alerts list

In the All Alerts list, the alert priority score value is displayed in the Priority column. For display purposes only, the actual calculated alert priority score is divided by 1000, while in the database, for calculation and sorting purposes, the full computed value is used. For example, for Alert0010002, the alert priority score displayed in the All Alert list is 20302, while in the Additional information tab, the actual value that is displayed in the Priority Breakdown field is 20302020.003, see the following images.

The detailed computation of the alert priority of the selected alert is displayed in the Priority Breakdown area Additional Information tab, as depicted in the image below.
Calculating the alert priority score

Priority is calculated for alerts that match these conditions:

- State not-equal to Closed
- Severity not-equal to Info

Alerts that are queued for priority calculation are listed in the `em_alert_trigger_queue` table. In the Scheduled Jobs table, there are two Event Management - Alert Priority Queue jobs. These two jobs can run in parallel. You can use Insert and Stay to create another job. The Event Management - Alert Priority Queue job runs once every minute and calculates priorities in batches of 1000 open alerts. There are multiple factors that determine the value of the priority of an alert. A change in severity triggers an immediate update of the alert priority score. In this case, only the severity-part of the priority is changed, together with severity information in the Priority Breakdown area of the Additional Information tab.

Triggers that cause recalculation

Triggers that can cause the alert priority score to be recalculated are:

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Effect on the alert priority score</th>
</tr>
</thead>
<tbody>
<tr>
<td>New alert</td>
<td>The generation of a new alert is a trigger to calculate the alert priority score.</td>
</tr>
</tbody>
</table>
### Trigger

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Effect on the alert priority score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existing alert that changes from closed to any other state</td>
<td>Priority is not calculated for a closed alert, so if the alert state is changed from its current state of closed to open, reopen, or flapping, that change is a trigger to recalculate the alert priority score.</td>
</tr>
<tr>
<td>Role change, between primary, secondary, or none</td>
<td>Changes in role are a trigger for the alert priority score to be recalculated. For example:</td>
</tr>
<tr>
<td></td>
<td>· An alert whose type has changed to Primary, triggers the alert priority score to be recalculated.</td>
</tr>
<tr>
<td></td>
<td>· An alert whose type has changed to None, triggers the alert priority score to be recalculated.</td>
</tr>
<tr>
<td></td>
<td>· An alert whose type has changed to Secondary, triggers the alert priority score to be recalculated.</td>
</tr>
<tr>
<td>CI type</td>
<td>When a CI becomes bound to alert, that change is a trigger to recalculate the alert priority score. This trigger to recalculate the priority occurs once only, when the CI is bound to the alert.</td>
</tr>
<tr>
<td>Change of number of secondary alerts</td>
<td>A change in the number of secondary alerts triggers the alert priority score to be recalculated.</td>
</tr>
<tr>
<td>Severity</td>
<td>A change in severity is a trigger to recalculate the alert priority score. For example, changing from Info to Major triggers the score to be recalculated. Unlike the other triggers listed above, a change in severity triggers an immediate update of the severity part of the alert priority score calculation.</td>
</tr>
</tbody>
</table>

### Modify the alert priority score

You can change the importance of some categories of the alert priority, by modifying their order and/or their weight, as described below. For example, if the CI type is higher in importance than the number of impacted business services, you can change their respective order. As a result, the number of business services is now multiplied by 100, while CI type is now multiplied by 1000000.

**Note:** The changes that you make, using this advanced procedure, changes the default method of calculating the alert priority score. Alerts that might otherwise not have a high score, by changing these configurable values, changes the way you determine the order in which to handle alerts.

### Changing category order

Navigate to the `em_alert_priority_category_order` table. In the **Order** column, you can change the order of the required category.

### Category value mapping

The `em_alert_priority_category_mapping` table shows the configuration value for each category choice. Each value in a drop-down list of categories can be remapped to a different value by configuring this table.

### CI type weight factor mapping

Navigate to the `em_alert_priority_ci_type` table. You can add a new CI, for example, `DualCoreCPU` and in the **Priority** column, you can change the priority of the required category.

In addition, you can edit existing Type and also change its priority value. This table is used to map the value of each CI type, for example, a mainframe that is CI_type: cmdb_ci_mainframe might
have a priority of 80, while a server with CI type: cmdb_ci_server might have a priority of 6 0. The mapping enables you to customize the priority of the various CI types.

**View discovered service history**
The discovered service history shows the frequency of discovered services for a particular time period.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

You can view the discovered services history by business service. The discovered services history is not available for application services. The discovered services history appears only when there are multiple discovered services that occur over a period. A discovered service is a business service that is discovered by Service Mapping. If no discovered services are generated for a particular business service, the discovered services history is hidden. The color corresponds to the discovered services severity, and the length of the bar in each color corresponds to how long the discovered services stayed at that severity.

**Discovered services history**

1. Navigate to **Event Management > Dashboard**.
2. Double-click a business service tile.
3. On the business service map, click a CI that has a severity color.
4. To change the information that appears on the discovered services history, click a history icon.

   - **Calendar icon**: Show information for currently active discovered services.
   - **H**: Show information for discovered services that occurred in the past hours.
   - **D**: Show information for discovered services that occurred in the past 24 hours.
   - **W**: Show information for discovered services that occurred in the past week.
   - **M**: Show information for discovered services that occurred in the past month.

**How alerts work with CIs in maintenance**

When a CI is in maintenance, the impact tree, the business service map, and Alerts tab are updated based on various factors.

A CI is in maintenance when:

- A change request is scheduled for the CI.
- The **Status** field on the CI record is set to **In Maintenance**.

**Note**: To customize how alerts work with CIs in maintenance, see **Create maintenance rules**.

You can learn about how alerts work with CIs from the following video tutorial.
How CIs in Maintenance appear in Event Management

<table>
<thead>
<tr>
<th>Description and the optimal time to resolve the alert</th>
</tr>
</thead>
<tbody>
<tr>
<td>An active change request has the following values:</td>
</tr>
<tr>
<td>• The State is Scheduled or Implement.</td>
</tr>
<tr>
<td>• The current time is between the Planned start date and Planned end date on the change request.</td>
</tr>
<tr>
<td>OR</td>
</tr>
<tr>
<td>• The current time is between the Actual start date and Actual end date on the change request.</td>
</tr>
<tr>
<td>You can monitor the progress of the change request. Wait until the change request moves to the Review or Closed state. Then you can address all alerts for the affected CI and any alerts that generated between start and end dates. Use the impact tree, topology, and Alerts tab to show the calculated impact severity.</td>
</tr>
</tbody>
</table>

Note: The Maintenance check box for an alert is selected when the Status field on the CI record is In Maintenance. This Maintenance check box indicates that the alert must be hidden from the Alerts tab and Alert console.

If an active change request is scheduled for the CI or if the Status of the CI is In Maintenance, all alerts on the affected CI are excluded from impact calculation. The Alerts tab and Alert console also temporarily hide all corresponding alerts. The impact tree shows the CI in green with a note of (In Maintenance). The impact tree and the business service map temporarily show CIs in green.

For a business service, all alerts on CIs in the business service are also hidden from the Alerts tab. The entire business service is shown in green on the impact tree. For a host with an active change request, the host applications are considered as one unit. All child applications are treated in the same manner as the host until the change request is no longer active.
How CIs in Maintenance appear in Event Management

Description and the optimal time to resolve the alert

An inactive change request has the following values:
- The State is New, Assess, Authorize, Review, Close, or has no changes at all.
- The State is Scheduled or Implement. However, the current time is before or after the planned or actual date ranges on the change request.

You can monitor the progress of the change request. When a CI has an inactive change request, you can address the corresponding alerts as appropriate.

CI with no active change request and not in maintenance

When there is no active change request for a CI and when the CI is not in maintenance, impact calculation resumes. The impact tree, the business service map, and Alerts tab show the calculated impact severity for alerts.

Alert similarity

Finding alerts that are similar to the alert that you are currently investigating can help save troubleshooting time by seeing how similar alerts were resolved.

Finding similar alerts

Alert similarity applies natural language processing (NLP) based on Now Platform machine learning (ML) as implemented in Predictive Intelligence.

ML logic filters out irrelevant information and provides further information to help you understand the cause of alerts. You can view solutions used to resolve similar alerts and the probable root cause.

ML helps find similar alerts based on text found in these alert record fields:
- Description
- Short description
- Source
- Type
- Resource
- Metric name
Additional alert information

In the Alert Workspace, select the required alert and click **Insight** to view additional information about the alert.

In the Alert Console, select the required alert and use the group of tabs on the right, the "alert insight tabs," in the alert record to view additional information and insight about the alert. For example, you can view the top recent alerts for the current CI and similar CIs to find the most suitable solution to remediate and resolve the alert.

The alert insight tabs are:

- Repeated Alerts
- Similar Alerts
- CI Incidents
- CI Change Requests
- CI Problems
- Related Incidents
- Related Change Requests
- Related Problems

For more information about the alert insight tabs, see [Alert insight information](#).

Find similar alerts

You can find alerts similar to the alert currently being investigated. Save troubleshooting time by reviewing similar alerts to see how they were resolved.

Ensure that the Predictive Intelligence plugin is activated. To activate this plugin, see [Activate a plugin](#).

Ensure that the `evt_mgmt.similarity_use_ml` property is set to `true`.

Role required: evt_mgmt_admin

Save time in resolving the current alert by reviewing how similar alerts were resolved. Provide resolution consistency by resolving similar alerts in the same way.

1. Navigate to **Event Management > All Alerts**.
2. In the list of alerts, click the alert record that you want to investigate.
3. In the alert record, click the **Similar Alerts** tab.
   *Alerts that are similar to the alert that is being investigated are listed.*
By default, up to 10 similar alerts are listed. You can modify this value by setting the `evt_mgmt.similarity_max_similar_alerts_shown` property.

4. If you configured a threshold, in the Related links area, click **Show Similarities Details**.

The Sys ID and confidence level of each similar alert is displayed. The Confidence level is an indication of how the similar alert matches the parameters that you set. As an example, a high confidence percentage value indicates a higher likelihood of similarity of the alerts to the values that you set.

You can modify the value of the threshold by setting the `evt_mgmt.similarity_use_threshold` property.

**Similarity solutions**

Similarity solutions enable you to use Machine Learning (ML) to compare the text in a resolved alert record to an open alert record to reuse its resolution approach.
Training a similarity solution

To train a similarity solution, you collect words to compile a collection that Machine Learning (ML) can use to compare text in the **Short Description**, **Description**, **Source**, **Type**, **Resource**, and **Metric Name** fields in a resolved alert to see whether the words in the set match words in an open alert. The resolved alert, which is similar to an open alert, provides an example to show how the open alert can be resolved.

To train a solution, the filter must return at least one record. If your filter returns no records, update it.

**Note:** The preferred number of records for training a solution is between 30,000 records and 300,000. If you submit more than 300,000 records, the most recent 300,000 records are used to train the solution. Use only authentic records from the database.

- Ensure that the records you train are not too old and that they are relevant to your business needs. Keep the words in the collection current.
- Do not use hard-coded dates as filters because these filters are not updated when you retrain solutions unless you update them manually before every retraining. Instead, use relative date filters, for example, the last 3 months, last 6 months, or last 12 months.
- Perform training as needed until it provides an acceptable similarity solution. This practice provides you time to review and update your solution definition.

Fields to include in the solution

Record the fields that are likely to contain words and phrases that help the system identify similar records for your solution.

The similarity fields that you select should be a subset of your input field selections. For example, if you select fields from incident records that are in Open state, do not select **Close note** as a similarity field. Because open records do not include **Close note** fields, the text cannot be similar.

The similarity fields are available to users when they create records.

About the similarity score

The similarity score is a measure from 0-100 of the degree of similarity between two alert records. Alert records that have a similarity score higher than the threshold that you specify is returned by the solution.

Review similarity examples and their scores using the **Show training progress** feature to determine whether to either increase or decrease the solution threshold. You can change the threshold value in the **Threshold for Similarity Score** field.

View training solution progress

Training times vary based on the number of records and classes within the training set. The more records and classes you use, the longer the training can take. For example, a data set containing 100,000 records and several hundred classes can take around five hours to complete.

To show the training solution progress, the ML solution automatically performs the following activities when you select **Show training progress** on the Solutions page. For more information, see **View solution training progress**.
Solution training activities

<table>
<thead>
<tr>
<th>Activity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fetching files for training.</td>
<td>The system downloads the training records and sends them to the nearest training service.</td>
</tr>
<tr>
<td>Preparing the data.</td>
<td>The system removes duplicate records from the training set.</td>
</tr>
<tr>
<td>Training the solution.</td>
<td>The training service trains the solution.</td>
</tr>
<tr>
<td>Uploading the trained solution.</td>
<td>The training service uploads the solution as attachment records.</td>
</tr>
</tbody>
</table>

Create an Event Management similarity solution
Create and train a solution that applies machine learning to a collection of words to target and suggest similar alerts in your instance dataset. For example, you can compare the text in a resolved alert record to an open alert record to reuse its resolution approach.

Role required: evt_mgmt_admin

The encryption scheme that you use can affect whether you can train solutions.
- If your data is encrypted by Full Disc Encryption (FDE), you can train solutions.
- If the field you are using for training is encrypted using Platform Encryption, ensure that the shared service worker user has the correct encryption context role that has been used for encryption.
- If you are using Edge Encryption, training is not supported.

Note: This feature supports only English language processing.

1. Navigate to Predictive Intelligence > Similarity > Solution Definitions.
2. Click New.
3. On the form, fill in the fields.

ML Solution Definition form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution Template</td>
<td>Use to define a word collection.</td>
</tr>
<tr>
<td>Label</td>
<td>Unique name for the similarity solution.</td>
</tr>
<tr>
<td>Name</td>
<td>Field that is auto-populated with a system-assigned name that is similar to the label name when the solution is created.</td>
</tr>
<tr>
<td>Table</td>
<td>Table that contains the records to use for the word collection, for example, the Incident table or the Asset table. When you assign a table value, a link appears that shows the number of records that match your current conditions.</td>
</tr>
<tr>
<td>Filter</td>
<td>Conditions that you want to apply to the training records. To train a solution, the filter must return at least one record. If your filter returns no records, update it.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Input Fields</td>
<td>Record the fields that contain the text and context that you want to include in your word collection. The field data type can be a string, reference, choice, or HTML, such as Short description and Description.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Journal type is not a supported data type. Good candidates for your input fields have text that is relevant to the solution. For example, if you are configuring a solution to find similar incident records, you can select the Short Description, Description, Resolution notes, and Close notes fields.</td>
</tr>
<tr>
<td>Training Frequency</td>
<td>Training is performed once by the system, by default. Because your data can age over time and degrade the accuracy of recommendations, consider invoking scheduled trainings once your solution definition is fairly stable.</td>
</tr>
<tr>
<td>Domain</td>
<td>On instances where domain separation is active, select the domain whose target records you want to use for your word collection. Create a separate similarity solution definition record for each domain whose field values you want to use for your word collection.</td>
</tr>
<tr>
<td>Similarity Fields</td>
<td>Record fields that are likely to contain words and phrases that help the system identify similar records for your solution. To change your Similarity Field choices, click the Lock icon ( ) to open the field and make your updates. Click the icon again to close the field and save your updates.</td>
</tr>
<tr>
<td>Similarity Window</td>
<td>The period in which you want to look for similar records. If you have a smaller number of total incoming records, a larger window might be best. However, if the window is too large, you may retrieve records that are not as useful. If the window is too small, you might not retrieve enough similar records. For example, if you are looking for similar incident records that are open, you can select Last 1 day. This selection targets the most recent records, many of which could still be open.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Similarity Window Filters</td>
<td>Filter conditions for your similarity window. These filters define the dataset conditions under which your similarity results are determined. For example, if you are looking for similar incident records that are open, you can filter your search by creating conditions such as: (Incident state) (is) (In progress) These filters are applied in addition to the Input Field filters.</td>
</tr>
<tr>
<td>Window Refresh Frequency</td>
<td>Frequency to refresh the similarity window. For example, if your window contains incident records that are open, you can select a refresh frequency value of Every 15 minutes. New incidents typically occur frequently throughout the day so this frequency increases the likelihood that newly opened records are included in the refresh. Note: If your similarity window is composed of records such as Knowledge article records, which are typically not created often, you can choose a larger refresh frequency such as Every 1 day.</td>
</tr>
</tbody>
</table>

4. Click **Save** to save the solution definition record so you can return to it later, or click **Submit & Train** to create the solution definition record and train it.

5. If you submitted the solution for training, click **OK** on the **Training Activation** window to confirm.

The system schedules the solution for training with the nearest training service. You and other users subscribed to the Predictive Intelligence Notifications category receive a notification when the training completes, including any errors that might have occurred in the training. When training is complete, the system uploads the solution as an Attachment record.

Review the trained similarity solution examples in the **Similarity Examples** related link on your ML Solution Definition form.

View Event Management solution training progress

View solution training progress or statistics to determine whether a solution is available or how long the next training cycle might take to complete.

Role required: evt_mgmt_admin

1. Navigate to **Predictive Intelligence > Similarity > Solutions**.
2. In the ML Solutions list, select the solution whose progress or statistics you want to view.
3. Click the **Show training progress** related link.

The ML solution automatically performs activities that appear in the Training Progress pop-up window.

The system shows a Training Progress pop-up window. In the following example, the progress failed.
If the training is a success, all bars would be fully blue and the progress for each step would show 100%.

See Review Event Management similarity examples. Review Event Management similarity examples
Review the similarity examples and scores that the system provides during solution training to see how the selected alert record compares to existing alert records. For example, you can modify the similarity score threshold to increase the accuracy of your similarity recommendations.

Train a similarity solution in a ServiceNow datacenter. For more information, see Create an Event Management similarity solution.

Role required: evt_mgmt_admin

Adjust the current threshold to provide useful results. For example, a low similarity score has more coverage but less precision.

1. Navigate to Predictive Intelligence > Similarity > Solutions.
2. In the ML Solutions list, locate your solution and click the Information icon (i).
3. Click Open Record.
4. Click the Similarity Examples related link.
5. Review the similarity examples and their threshold scores to determine the accuracy and coverage levels you want applied to your solution similarity results. The higher the similarity score, the more precise it is and the less coverage it offers. The lower the similarity score, the more coverage it has and the less precision it offers.

6. Based on your review, determine whether to increase or decrease the similarity score threshold value for your similarity solution. If the current threshold is not providing useful results, adjust the threshold value.

7. Navigate to Predictive Intelligence > Similarity > Solutions.

8. In the ML Solutions list, locate your solution and click the Reference icon.

9. Click Open Record.

10. In the Solution Statistics tab, enter a new value in the Similarity Score Threshold field. For example, suppose that the current score is 80. Based on your similarity example review, you might want to update the field by entering a higher score of 90, which will return fewer results but ones that are more similar to the alert record under review. A lower similarity score has more coverage but less precision.

11. Right-click in the form header and click Save. Your solution uses the new similarity threshold value. In the example, the solution returns similar results that have a score higher than 90.

Activate Event Management solution version
The system activates the most recent version of the solution, but you can activate any previously trained Event Management solution version if it is more appropriate.

As you might have created more than one solution, activate the solution that is most appropriate.

Role required: evt_mgmt_admin

1. Navigate to Predictive Intelligence > Similarity > Solutions.

2. In the ML Solutions list, click the Reference icon for the trained solution that you want to activate.

3. Click Open Record. You can assign a title to the solution using the evt_mgmt_similarity_solution_title property. The default value is ml_x_global_alert_similarity. This property changes the value in the Solution Name field.

4. In the solution record, click Activate.
The system activates this solution version and deactivates any other solution version.

Alert and CI remediation

Alert and configuration item (CI) remediations help troubleshoot and resolve underlying problems that generate alerts. Remediation is based on Orchestration workflows that can be scripted to perform remediation tasks such as gathering system information or rebooting a server.

Event Management provides a framework for configuring and implementing a system-wide remediation plan to resolve, minimize, or prevent problems that generate events and alerts. Configure alert management rules to remediate alerts and CI remediation rules to remediate a set of CIs.

Alert remediation

Use alert management rules to associate specific alerts with a remediation workflow. You can apply a workflow to alerts manually. Alternatively, you can configure the alert management rule that executes a workflow automatically for alerts that match the rule criteria. The workflow can also be scripted to request user approval before executing subsequent workflow tasks. The workflow is paused for user approval, even if it was automatically triggered.

CI remediation

Configure remediation more comprehensively by creating CI remediation rules to manually apply remediation to specific CIs.

Remediation tasks

When a remediation workflow is executed, a remediation task is created to capture details such as the time that the workflow started to run, the alert that triggered the remediation workflow (if relevant), and the CI that was remediated. View these tasks to track remediation activities in the organization.

Apply alert remediation

If an alert meets the criteria of an alert action rule that enables manual remediation, you can manually apply remediation to resolve the alert.

Ensure that the alert matches an alert action rule that enables manual remediation.

Role required: evt_mgmt_admin

When an alert is initially created, you can select which remediation to run from a list of all applicable remediations.

1. Find the alert that you want to remediate.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| From the Event Management dashboard         | Navigate to Event Management > Dashboard. You have these options for selecting an alert:  
  - Select an alert from the list at the bottom of the dashboard.  
  - Click a business service tile and select an alert from the list at the bottom of the service map.  
  - Click a technical service tile and select an alert from the technical service object list, or select an alert from the Alerts tab.                                                                                                                                                                                                                                                                                                                                                             |
| From the Alerts Console                     | 1. Navigate to Event Management > Alerts Console.  
  2. Select an alert from the list.  
  2. Right-click the alert, and select Quick Response.  
  3. In the Quick Response dialog box, select the link to the remediation workflow that you want to apply.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

Create or edit CI remediation
Create a CI remediation rule that lets users manually apply an Orchestration workflow for resolving issues with specific CIs associated with alerts. Define these CIs in the CI filter conditions of the rule.

Create the workflow to remediate CIs. In the workflow settings, select Remediation Task (em_remediation_task) in the Table field.

Then submit the workflow definition, and add the following conditions to the workflow properties:

- (Run Remediation)(is)(true)
- (Workflow)(is)(&lt;name of the new workflow that you just created&gt;)
  For example, if the name of the workflow is CI remediation, add this condition:
  
  (Workflow)(is)(CI remediation)

After you finish configuring the workflow, make sure that you publish it. For more details, see Create a workflow.

Role required: evt_mgmt_admin

A CI remediation rule associates a set of CIs that might experience problems with a remediation workflow. The remediation workflow can either resolve the underlying problem or help troubleshoot the problem that generated the alert. For example, you can proactively configure a workflow with the appropriate response actions for predictable alerts. For more details, see the Event Management Remediation short video.

2. Click New, or select a CI remediation to edit.  
3. Fill in the fields, and click Submit or Update.

<table>
<thead>
<tr>
<th>Control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI filter</td>
<td>Conditions to match the CIs that you want to apply this remediation to.</td>
</tr>
</tbody>
</table>
In service maps that are opened from the Event Management dashboard, this remediation can be applied to any CIs that match the filter conditions. For more information, see Apply CI remediation.

**Apply CI remediation**

You can apply remediation to a CI that matches the conditions in a CI remediation rule. The specified remediation workflow is then executed to perform tasks that help resolve or troubleshoot a problem that the CI is experiencing.

Ensure that the CI matches a CI remediation rule.

Role required: evt_mgmt_admin

1. Navigate to Event Management > Dashboard.
2. Double-click the tile of the service that includes the CI.
3. On the service map, right-click the CI and select Remediation options.
4. In the Remediation options dialog box, select the remediation that you want to apply and click Run.

The remediations list includes remediations in which the CI matches the filter of a CI remediation rule, and remediations configured in alert action rules for which alerts that are associated with the CI, match the filter.

**View remediation tasks**

Event Management automatically creates a remediation task to capture every remediation that was applied to a CI or to an alert. It gives you an overall view of remediation activities in the organization.

Role required: evt_mgmt_user

Navigate to Event Management > Remediation Tasks.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>Number assigned to this task.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>Item or service affected by this task.</td>
</tr>
<tr>
<td>Priority</td>
<td>Priority that was assigned to this task, based on impact and urgency.</td>
</tr>
<tr>
<td>State</td>
<td>State of the task, which can be open or closed.</td>
</tr>
<tr>
<td>Assigned to</td>
<td>Person that this task was assigned to.</td>
</tr>
<tr>
<td>Short description</td>
<td>Description for the task.</td>
</tr>
<tr>
<td>Task type</td>
<td>This task type is a remediation task.</td>
</tr>
</tbody>
</table>

**Use the Event Management overview dashboard**

The Event Management overview module uses Performance Analytics to present data from your instance for you to better visualize and understand your processes and drive continual improvement.
The default number of rows that are allowed to be fetched from an Indicator Source is 50,000. To modify this number, edit the com.snc.pa.dc.max_row_count_indicator_source property that is located here: **Performance Analytics > System > Properties**.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

With the Event Management administration role, you can add, delete, and rearrange report widgets. The other Event Management roles can view and refresh report widgets.

Note: Event Management does not support mobile devices.

1. Navigate to **Event Management > Overview**.
   
   The default display of the Event Management Overview dashboard depends on whether the base instance is an upgrade from an earlier release and which plugins are activated:
   
   - Event Management upgraded from an earlier release or is activated in the instance, see **Event Management Overview dashboard**.
   - Event Management and the Performance Analytics - Content Pack - Event Management (com.snc.pa.em) plugin are activated, see **Event Management Insight dashboard**.

   Note: If Event Management is upgraded from an earlier release and Performance Analytics - Content Pack - Event Management (com.snc.pa.em) plugin is also activated, see **Event Management Scorecards**.

2. Move or add widgets as needed.
3. Click elements within widgets to obtain more information.

**Event Management Insight dashboard**

The Event Management Insight dashboard is available to be displayed when both the Event Management plugin (com.glideapp.itom.snac) and the Performance Analytics - Content Pack - Event Management (com.snc.pa.em) plugin are activated.

**Event Management Insights dashboard**

Note: Event Management does not support mobile devices.

1. Navigate to **Event Management > Overview** to view the overview module.
2. Select Event Management Insights from the drop-down list.

The tabs available in the Event Management Insights dashboard are:

- Data Quality
- Services Criticality Monitor
- Impact Time Severity Monitor
- Alert Lifecycle

You can click elements within widgets to obtain more information.

**Data Quality**

The Data Quality tab provided in the base instance displays:

- Open Alerts older than 7 days by severity and state
- Open alerts not updated in the last 5 days by severity and state
- Open alerts where related INT is not updated in the last 3 days
- % of alerts in un-acknowledge
- % of un-grouped alerts
- % of alerts without assigned task
Open Alerts older than 7 days by severity and state

<table>
<thead>
<tr>
<th>Severity</th>
<th>Open</th>
<th>Reopen</th>
<th>Flapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>18</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Major</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Minor</td>
<td>6</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Warning</td>
<td>5</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Info</td>
<td>7</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

Open alerts where related INT is not updated in the last 3 days

<table>
<thead>
<tr>
<th>Severity</th>
<th>Open</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>18</td>
</tr>
<tr>
<td>Major</td>
<td>1</td>
</tr>
<tr>
<td>Minor</td>
<td>1</td>
</tr>
<tr>
<td>Warning</td>
<td>2</td>
</tr>
<tr>
<td>Info</td>
<td></td>
</tr>
</tbody>
</table>
Services Criticality Monitor

The Services Criticality Monitor tab provided in the base instance shows the Impact Time by Services Criticality grouped by:

- Most Critical
- Somewhat Critical
- Less Critical
- Not Critical
Impact Time by Services Criticality

Most Critical: 290

Sep 26 -
Most Critical
290 Mins

% Impact Time of Today
20.12

% of Total Impacted Services
17.58

Breakdowns

Most Critical, Sep 26: 290

Auto Business Services ▼ Select an Element ▼

Name

Blackberry

58 Mins
Impact Time Severity Monitor

The Impact Time Severity Monitor tab shows the Impact Time by Severity grouped by:

- Critical
- Major
- Minor
- Warning
Impact Time by Serverity

Critical: 398
Major: 56

Sep 26

Critical

398 Mins

Breakdowns

Auto Business Services
Select an Element

Name
ServiceNow Event Management
Al_group
Blackberry

Sep 26
80 Mins
52 Mins
48 Mins
Alert Lifecycle

The Alert Lifecycle tab shows the Alert Lifecycle widgets grouped by:

- New
- Open
- Closed

Within each of these widgets, you can display either Breakdowns or the underlying Records.
### Alert Lifecycle

**New: 54**  
**Open: 77**

#### Sep 26

**New**

54

#### % of new critical alerts

63

#### Breakdowns

<table>
<thead>
<tr>
<th>Name</th>
<th>Sep 26</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Closed</td>
<td>39</td>
<td></td>
</tr>
</tbody>
</table>
Event Management Scorecards
The Event Management Scorecards dashboard is available to be displayed when both Event Management plugin (com.glideapp.itom.snac) and the Performance Analytics - Content Pack - Event Management (com.snc.pa.em) plugin are activated.

Event Management Scorecards dashboard

**Note:** Event Management does not support mobile devices.

1. Navigate to **Event Management > Overview** to view the overview module.
2. Select Event Management Scorecards from the drop-down list.

The tabs available in the Event Management Scorecards dashboard are:
- EM Overview
- Service Scorecards
- Alerts Scorecard
- Health Check Monitors

You can click elements within widgets to obtain more information.

**EM Overview**
The EM Overview tab shows:

<table>
<thead>
<tr>
<th>Widget</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Alerts</td>
<td>Number of active alerts.</td>
</tr>
<tr>
<td>Critical Alerts</td>
<td>Number of alerts whose severity is critical.</td>
</tr>
<tr>
<td>Affected Business Services</td>
<td>Number of services that have an impact on the business service.</td>
</tr>
<tr>
<td>Affected Business Services - Grouped</td>
<td>Graphical representation of the number of services that have an impact on the business service. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Active Alerts - Grouped</td>
<td>Graphical representation of the number of active alerts. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Business Services Affected by Alerts</td>
<td>Number of business services that have an impact on alerts. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Active Alerts with Tasks</td>
<td>Number of active alerts that have tasks. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Top 10 Active Alerts</td>
<td>Ten most active alerts during the review period. You can configure the graph using the controls in the widget.</td>
</tr>
</tbody>
</table>
Active Alerts: 1,780
Critical Alerts: 596
Service Scorecard

The Service Scorecard tab shows:

**Default reports in the Service Scorecard dashboard**

<table>
<thead>
<tr>
<th>Widget</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Impacted Services</td>
<td>Displays the number of services that have been impacted by alert.</td>
</tr>
<tr>
<td>Average Services Impact Time</td>
<td>Displays the average time (in minutes) that services are impacted by alerts.</td>
</tr>
<tr>
<td>% Today's Affected Business Services</td>
<td>Displays the percentage of business services that were affecting during the day.</td>
</tr>
<tr>
<td>% Business Services Impact Time Today</td>
<td>Displays the percentage of business services that were impacted by alerts during the day.</td>
</tr>
<tr>
<td>Average Impact Time by Business Services</td>
<td>Displays the average time (in minutes) that business services are impacted by alerts.</td>
</tr>
<tr>
<td>Average Impact Time by Services Criticality</td>
<td>Displays the average time (in minutes) that business services are impacted by service criticality.</td>
</tr>
</tbody>
</table>
New Impacted Services

April 13

17

0 (0.0%) April 12: 17

% Today’s Affected Business Services

April 13

54.84%

0.00 (0.0%) April 12: 54.84%
Alerts Scorecard

The Alerts Scorecard tab shows:

- Average age open alerts
- % of alerts reopened at least once
- % of alerts closed same day opened
- % of primary alerts out of total alerts
- % of alert grouping compression
- % of new critical alerts
- % of alert grouping coverage
- % of alerts closed by operator
- Alert Group Feedback
**Health Check Monitors**

The Health Check Monitors tab shows:

- License usage per type
- Event status breakdown at the end of last 2 hours
- Number of events processed per job during the last 2 hours
Note: If there are a large number of events, the Event status breakdown at the end of the last 2 hours report is not available.

Event Management Overview dashboard
The Event Management Overview dashboard is available to be displayed either when Event Management is upgraded from an earlier release or activated in the instance for the first time.

Event Management Overview dashboard

Note: Event Management does not support mobile devices.

Navigate to Event Management > Overview to view the overview module.
You can click elements within widgets to obtain more information.
The overview modules that are available in this dashboard are:
- Event Management Overview
- Health Check Monitors

Event Management Overview tab
Default widgets in the Event Management Overview tab are:

<table>
<thead>
<tr>
<th>Widget</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Alerts</td>
<td>Displays the number of active alerts.</td>
</tr>
<tr>
<td>Critical Alerts</td>
<td>Displays the number of alerts whose severity is critical.</td>
</tr>
<tr>
<td>Affected Business Services</td>
<td>Displays the number of services that have an impact on the business service.</td>
</tr>
<tr>
<td>Affected Business Services - Grouped</td>
<td>Graphical representation of the number of services that have an impact on the business service. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Active Alerts - Grouped</td>
<td>Graphical representation of the number of active alerts. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Business Services Affected by Alerts</td>
<td>Displays the number of business services that have an impact on alerts. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Active Alerts with Tasks</td>
<td>Displays the number of active alerts that have tasks. You can configure the graph using the controls in the widget.</td>
</tr>
<tr>
<td>Top 10 Active Alerts</td>
<td>Displays the 10 alerts that were the most active during the review period. You can configure the graph using the controls in the widget.</td>
</tr>
</tbody>
</table>
### Health Check Monitors tab

Default widgets in the Health Check Monitors tab are:

<table>
<thead>
<tr>
<th>Widget</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>License usage per type</td>
<td>Displays the number of nodes that were counted for license usage.</td>
</tr>
<tr>
<td>Event status breakdown at the end of last 2 hours</td>
<td>Displays the status of events during the past 2 hours.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If there are a large number of events, this report is not available.</td>
</tr>
<tr>
<td>Number of events processed per job during last 2 hours</td>
<td>Displays the number of events processed per job during last 2 hours.</td>
</tr>
</tbody>
</table>
Self-health monitors for Event Management

Use the Event Management self-health monitors to track Event Management features and resolve issues.

Monitor Event Management to detect and resolve issues

Enable the self-health feature to help detect and troubleshoot Event Management health issues. Some configuration is required before the self-health feature can display meaningful data. Once configured and the related scheduled jobs are enabled, the Event Management dashboard displays data that is automatically collected and calculated on a recurring schedule.

The self-health feature creates CIs in the CMDB to represent the components of the self-health service. Use the provided monitors to monitor and track the health of business services and Event Management features.

Event Management self-health can:

- Trigger alerts when there is an issue with Event Management.
- Display the ServiceNow Event Management application service on the Event Management dashboard. Use the dashboard to view alerts by business service, application service, technical service, and alert groups. You can double-click a tile to view a map representation of the service.
- View a map representation of the ServiceNow Event Management application service to see the relationships between CIs in the ServiceNow Event Management application service.
Set up

Ensure that the Event Management — self-health monitoring (evt_mgmt.self_health_active) property is set to Enable to use the Event Management features that handle issues with the Event Management application itself. For more information, see Start or stop self-health monitoring.

Self-health monitoring process workflow

After the Event Management — self-health monitoring property is enabled:
1. The **ServiceNow Event Management** application service is automatically created.
2. Jobs are automatically scheduled to run scripts that configure what must be monitored.
3. The Monitoring States (em_monitor_state) table is updated after each event run that the configuration scripts trigger.
4. Monitor the alerts that are generated. The business service displays the status of the monitored entity. For example, alerts are displayed in the Event Management dashboard or Alert Console.
5. Resolve the alerts according to the underlying issue, for example, using **Quick Response**.

### Jobs scheduled to process scripts

The jobs scheduled to run by default after the **Event Management — self-health monitoring** property is enabled are:

- The **Event Management - Insert Health Monitor** scheduled job which runs the scripts. Determine what must be monitored by adding to, or modifying the script in this job which runs once every hour. After the job has run, you can view the **ServiceNow Event Management** application service.

- The **Event Management - Update Health Monitor** scheduled job which runs the update scripts. This job runs once every minute.

### Start or stop self-health monitoring

You can control the starting or stopping of the self-health monitor feature by configuring the self-health monitoring property. The first time that the self-health monitoring property is enabled, it automatically creates the **ServiceNow Event Management** application service.

**Role required:** evt_mgmt_admin

Configure the self-health monitoring property to monitor and track application services and Event Management.

1. Navigate to **Event Management > Settings > Properties**.
2. Locate the **Enable Event Management self-health monitoring (evt_mgmt.self_health_active)** property.
   Select one of these values:
### Enable Event Management self-health monitoring property

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>The ServiceNow Event Management application service is created and set to monitor self-health. Events are received and alerts are generated.</td>
</tr>
<tr>
<td>Alerts Only</td>
<td>The ServiceNow Event Management application service and the CIs related to it are not created. Alerts are created but are not bound to a CI.</td>
</tr>
<tr>
<td>Disabled</td>
<td>The ServiceNow Event Management application service is not operational. Events are not received, and alerts are not generated.</td>
</tr>
</tbody>
</table>

3. **Click Save.**

The ServiceNow Event Management application service is set according to the value selected for the `evt_mgmt.self_health_active` property.

**Monitor self-health with domain separation**

Use domain separation to enable self-health to display Event Management health issues that are based on data, rules, and settings from the logged-on user domain. The selected domain must not contain any child domains.

Ensure that the Domain Support — Domain Extension Installer (com.glide.domain.msp_extensions.installer) plugin is activated and that the MID Server you are connecting to is configured for Event Management.

**Role required:** `evt_mgmt_admin`

Self-health monitoring is domain aware. If the Domain Support — Domain Extension Installer (com.glide.domain.msp_extensions.installer) plugin has been activated, then self-health displays health issues based on data, rules, and settings from the logged-on user domain. If rules and settings are not defined for a child domain, then the settings of the parent are applied, recursively.

When the Domain Support — Domain Extension Installer (com.glide.domain.msp_extensions.installer) plugin is not activated, adding, or removing a connector or MID Server is immediately reflected in the map.

When the Domain Support — Domain Extension Installer (com.glide.domain.msp_extensions.installer) plugin is activated, updates are done periodically according to the interval specifications of the Event Management — Insert Health Monitor scheduled job.

1. Navigate to **System Scheduler > Scheduled Jobs**.
2. In the **Schedule** list, locate and select the Event Management — Insert Health Monitor job.
3. In the Event Management — Insert Health Monitor job, under Related Links, click **Configure Job Definition**.
4. Set the required periodic update time in the **Repeat Interval** field by specifying the required values. The default interval is one hour.
5. Click Update.

Further configuration is required for CI creation, receipt of events, and alert generation. The `sys_id` of the domain must be specified in the Value field of the `evt_mgmt.domain_self_monitoring` property.

1. Navigate to System Properties All Properties.
2. Locate and select the `evt_mgmt.domain_self_monitoring` property.
3. In the Value field, specify the `sys_id` of the domain.
4. Click Update.

Configure a self-health monitor

You can configure a self-health monitor to track Event Management components and see that they do not exceed the specified threshold.

Role required: `evt_mgmt_admin`
The Monitoring Configurations (em_monitor_conf) table lists the records that determine which Event Management components are being monitored. Each record contains the health monitor script that is relevant to what you are monitoring. If you intend to create a new health monitor script, you must configure a monitor to match the purpose of the custom script. You can configure the monitoring configuration records to track the items that are important to you. For example, you can modify the threshold aspect in the Connector’s Idle state monitoring record, by specifying that the monitoring job frequency is 90 seconds instead of 120 seconds.

**Note:** Each of the monitoring configuration records can be disabled.

If a monitoring configuration specifies a threshold, the configuration value determines what alert severity to display. Threshold values specify the delay time values in units of minutes.

**Note:** Additional self-health information appears on the Event Management Overview dashboard.

After the Event Management - self-health monitoring property is enabled, these jobs are automatically scheduled to run scripts that configure what must be monitored:

- **The Event Management - Insert Health Monitor** scheduled job runs the scripts. Determine what must be monitored by adding or modifying the script in this job, which runs once every hour.
- **The Event Management - Update Health Monitor** scheduled job runs the update scripts. This job runs once every minute.

Use the following procedure to configure whether a monitoring configuration can run and set the frequency of the monitoring job. You can specify the threshold values that, if exceeded, trigger an alert.

1. **Navigate to Event Management > Settings > Self-Health configuration.**
2. From the Monitoring Configurations list, select a monitoring configuration record and modify the required field. On the form, fill in the fields.

### Monitoring Configuration form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Check box for enabling the monitoring configuration to be active.</td>
</tr>
<tr>
<td>Monitoring job frequency (seconds)</td>
<td>Frequency that the monitoring job is run.</td>
</tr>
<tr>
<td>Critical Threshold</td>
<td>Any point above the value is considered a critical consequence.</td>
</tr>
<tr>
<td>Major Threshold</td>
<td>Any point above the value, but below the value of the Critical Threshold, is considered a major consequence.</td>
</tr>
<tr>
<td>Minor Threshold</td>
<td>If exceeded, the value is considered a minor consequence.</td>
</tr>
<tr>
<td>Warning Threshold</td>
<td>If the value is exceeded, a warning is issued.</td>
</tr>
</tbody>
</table>

3. **Click Update.**

The values in the modified monitoring configuration record are compared to the actual values as they occur in events that are received. Alerts are created according to the configuration settings.
Example 1: Use the following procedure to modify the Critical Threshold field of the Connector’s Idle state monitoring monitor from 30 seconds to 35 seconds. This field determines the number of seconds that pass before an alert is generated when the Critical Threshold idle state value of a connector is exceeded. In this example, the Critical Threshold field is modified from 30 seconds to 35 seconds.

1. Navigate to Event Management > Settings > Self-Health configuration.
2. From the Monitoring Configurations list, select Connector’s Idle state monitoring.
3. In the Critical Threshold field, modify the value to 35.
4. Click Update.

Result The threshold value of the Critical Threshold field is increased from 30 to 35, thereby increasing the number of seconds that pass before an alert is generated for this threshold.

Example 2: This example describes the configuration of the severity threshold when using the Connectors Status monitor. The Connectors Status monitor has a true (= value 1) or false (= value 0) configuration for the severity threshold. For this monitor, where a threshold has the value 1, then that severity is displayed. Only one severity threshold can have the value 1. This example assumes that the required severity displayed is Minor.

1. Navigate to Event Management > Settings > Self-Health configuration.
2. From the Monitoring Configurations list, select Connectors Status.
3. In the Minor Threshold field, specify the value 1.
4. Click Update.

Result The monitor issues an alert if the threshold value of the Minor Threshold field is exceeded.

Create a customized monitor. A template is provided to assist you to create script that can monitor the specified components. For more information, see Create a self-health monitor to use custom health monitor script.

Create a self-health monitor to use custom health monitor script
You can create a self-health monitor to use custom health monitor script to monitor specified Event Management components.

Role required: evt_mgmt_admin

To augment the self-health monitors that are provided with the base system, you can create a self-health monitor to monitor the components that you specify. The health monitor scripts that are provided with the base system are listed in the following table.

Health monitor scripts provided with the base system

<table>
<thead>
<tr>
<th>Health monitor script</th>
<th>Used by a monitor to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check delay in event processing</td>
<td>Monitor the duration of events (in minutes) that remain in Ready state and are not processed.</td>
</tr>
</tbody>
</table>
### Health monitor script

<table>
<thead>
<tr>
<th>Health monitor script</th>
<th>Used by a monitor to:</th>
</tr>
</thead>
</table>
| Connectors Monitor    | Monitor:  
  - Whether the time that a connector in idle state surpassed the threshold (in minutes) configured.  
  - The active status of connectors. |
| Get Event Processing state | Monitor the time that events are in Ready state. Alerts are sent when this time exceeds the threshold. |
| MID Server Threshold Alerts | Monitor MID Server health. Events are sent when changes in MID Server state are detected. |

You can create alert management rules to respond to and remediate the issues detected by self-health. Alerts are generated based on the specified severity and threshold. Navigate to **Event Management > Settings > Self-Health configuration** for the monitors that are provided with the base instance to monitor the issues detected by self-health, which are:

<table>
<thead>
<tr>
<th>Monitor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connector’s idle state monitoring</td>
<td>Monitor to verify whether any of the connectors were in idle state that surpassed the threshold (in minutes) that was configured. Health Monitor script selected: Connectors Monitor.</td>
</tr>
<tr>
<td>Connectors Status</td>
<td>Monitor to track the active status of the connectors. Select only one threshold to configure which severity to invoke using 1 to select the threshold field or 0 to ignore the threshold field. For more information, see Connector status example. Health Monitor script selected: Connectors Monitor.</td>
</tr>
<tr>
<td>Delay in event processing</td>
<td>Monitor to track the duration of events (in minutes) that remain in Ready state and were not processed. This duration period initiates an alert for the event processing CI with severity according to the configured thresholds (minutes). Health Monitor script selected: Check delay in event processing.</td>
</tr>
<tr>
<td>Event Processing job</td>
<td>Monitor the state of the event processing jobs. An alert is generated when the idle time of a job exceeds the configured threshold. Health Monitor script selected: Get Event Processing state.</td>
</tr>
<tr>
<td>MID Server Threshold Alerts</td>
<td>Monitor MID Server health. Events are sent when issues with MID Server availability and resource thresholds are detected. Health Monitor script selected: MID Server Threshold Alerts.</td>
</tr>
</tbody>
</table>

**Note:** To prevent a self-health monitor from detecting the connector status, at any time you can clear the **Active** check box of the monitor and then click **Update**.

1. Navigate to **Event Management > Settings > Self-Health configuration**, and click **New**.  
2. On the form, fill in the fields.
### Monitoring Configuration form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive and meaningful name for the monitor.</td>
</tr>
<tr>
<td>Description</td>
<td>Text narrative that describes the monitor.</td>
</tr>
<tr>
<td>Active</td>
<td>Check box to enable the monitor.</td>
</tr>
<tr>
<td>Monitoring job frequency (seconds)</td>
<td>Frequency that the monitoring job should run. Default: 120 seconds.</td>
</tr>
<tr>
<td>Last Run</td>
<td>Last run date and time. These values are automatically updated.</td>
</tr>
<tr>
<td>Script</td>
<td>Monitoring script. Click the search icon. The existing scripts are listed in the Health Monitor Scripts screen, or click <strong>New</strong> to create a custom health monitor script. For more information about creating a custom health monitor script, see step 3.</td>
</tr>
</tbody>
</table>

**Note:** Regarding all the following threshold configuration fields: All monitored values are compared against the thresholds. If a monitored value exceeds a threshold, an alert with the appropriate severity is issued.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical Threshold</td>
<td>Any point above the value is considered a critical consequence.</td>
</tr>
<tr>
<td>Major Threshold</td>
<td>Any point above the value, but below the value of the Critical Threshold, is considered a major consequence.</td>
</tr>
<tr>
<td>Minor Threshold</td>
<td>If exceeded, the value is considered a minor consequence.</td>
</tr>
<tr>
<td>Warning Threshold</td>
<td>If the value is exceeded, a warning is issued.</td>
</tr>
<tr>
<td>Additional information</td>
<td>Further information about the monitor.</td>
</tr>
</tbody>
</table>

3. To create a custom health monitor script, on the Monitoring Configuration form, next to the **Script** field, click the search icon to open the Health Monitor Scripts form, and click **New**.

4. On the form, fill in the fields.

### Health Monitor Script form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Check box to enable the monitor.</td>
</tr>
<tr>
<td>Description</td>
<td>Text narrative that describes the custom script.</td>
</tr>
<tr>
<td>Name</td>
<td>Descriptive and meaningful name for the custom script.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Script</td>
<td>Custom script to run when called from a monitor. A script template is provided as a guide.</td>
</tr>
<tr>
<td></td>
<td>You can use the <a href="https://service-now.com">Health monitor scripts provided with the base system</a> as examples of how to author the script, and use the <a href="https://service-now.com">script editor tools</a>.</td>
</tr>
<tr>
<td></td>
<td>Ensure that the custom script includes:</td>
</tr>
<tr>
<td></td>
<td>Run the monitor function, which is called according to the “monitoring job frequency” setting runUpdateHealthMonitorScripts(scriptId). Add this function to run the monitoring logic.</td>
</tr>
<tr>
<td></td>
<td>Specify the configuration function that is called every hour runInsertHealthMonitorScripts(scriptId). Add the configuration logic and add a CI if needed. In the navigation pane search field, enter em_monitor_state.list. In the Monitoring States (em_monitor_state) table, specify the initialization of the monitor record in the required record.</td>
</tr>
<tr>
<td></td>
<td>Use the helper functions defined in EvtMgmtHealthMonitorCommon and EvtMgmtEventAlertGenerator script includes:</td>
</tr>
<tr>
<td></td>
<td>· EvtMgmtHealthMonitorCommon</td>
</tr>
<tr>
<td></td>
<td>· healthMonitorCommon.shouldRun</td>
</tr>
<tr>
<td></td>
<td>· healthMonitorCommon.calculateSeverity</td>
</tr>
<tr>
<td></td>
<td>· EvtMgmtEventAlertGenerator</td>
</tr>
<tr>
<td></td>
<td>· openAlert</td>
</tr>
<tr>
<td></td>
<td>· closeAlert</td>
</tr>
</tbody>
</table>

You can use the Monitoring Configurations (em_monitor_conf) table for self-health monitoring configuration, the Health Monitor Scripts (em_monitor_scripts) table for scripts, and Monitoring States (em_monitor_state) table for displaying the results of the monitor, as initialized inside runInsertHealthMonitorScripts.

For more information about script includes and APIs, see:
- Script includes that are installed with Event Management
- [APIs and scripts](https://service-now.com)

5. **Click Submit.**

To monitor the Event Management components that you specify, you can configure a self-health monitor that uses the custom health monitor script that you created.

*View the Event Management self-health application service map*

You can view Event Management application service maps to have a visualization of the data on configuration items (CIs) that comprise this service, and the relations and connections between these CIs.

Role required: evt_mgmt_admin
The Event Management application service topology map only displays MID Servers that have passed validation and connectors that are active.

The Event Management components that are monitored by the self-health application service are listed in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>What is monitored</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event connectors</td>
<td>For each connector:</td>
</tr>
<tr>
<td></td>
<td>· Status: Success or Error</td>
</tr>
<tr>
<td></td>
<td>· Delay: Elapsed time from ‘last run time’ (above threshold)</td>
</tr>
<tr>
<td>Event processing jobs</td>
<td>For each event processing job:</td>
</tr>
<tr>
<td></td>
<td>· Number of actual jobs executed compared to the number of jobs configured to run.</td>
</tr>
<tr>
<td></td>
<td>· Delay of any of the event processing jobs above the time set as threshold.</td>
</tr>
<tr>
<td></td>
<td>· Duration of events in Ready state compared to the specified threshold.</td>
</tr>
<tr>
<td></td>
<td>· Delay of event rule processing above the set threshold. If there is a delay, an alert is generated and the event rule is disabled.</td>
</tr>
<tr>
<td>MID Servers</td>
<td>MID Servers that have been validated are displayed. The MID Server status and the Ecc_agent_issue table are monitored. An event is triggered whenever this table is updated. The MID Server status is displayed, either Up or Down.</td>
</tr>
</tbody>
</table>

For information about MID Server health, see [MID Server heartbeat](#).

1. Navigate to Event Management > Dashboard.
2. On the Event Management dashboard, double-click the ServiceNow Event Management application service tile, or navigate to Event Management > Services > Application Services.

3. In the application service screen, locate ServiceNow Event Management and click View Service.

Displayed in the map are the related MID Servers and the Event Management object, as depicted in the following image; these MID Servers have an Up status. MID Servers are updated when they are created or deactivated.
Under the Event Management object is Event Processing, Alert Processing, and Event Sources (for example, connectors). Only **Active** connectors are shown; when a connector is created, the connector is displayed only after it has been activated. The status of active connectors is updated and the status of non-active connectors is not displayed.

In the preceding image, the health of Event Management components is displayed. The omniConnector component has a red bar, indicating that a critical alert has occurred that affects this connector. In addition, this critical alert impacts the parent Alert Processing object, which as a result, has a red bar.

You can select the configuration item that has a critical alert and change the impact from parent to business service.
View event processing statistics

Extract statistics from your instance to ensure that performance is not affected and extract metrics related to event processes to monitor event processing status.

Ensure that the Event Management - Collect xmlstats job is active. Configure this scheduled job by navigating to System Definition > Scheduled Jobs.

You can control the processing of event statistics using these properties:

- `evt_mgmt.enable_event_processing_stats`. For information about configuring this property, see Enable processing of event process statistics.
- `evt_mgmt.event_processing_stats_period`. For information about configuring this property, see Configure statistics processing period.

Role required: evt_mgmt_operator

You can view key metrics from the instance to ensure that performance is not affected and monitor the status of the processing of events. For more information about viewing self-health monitoring statistics, see View self-health monitoring statistics for Operational Intelligence.

1. In a browser where the instance is running, run the XML Stats script by appending the following text to the instance URL: `xmlstats.do?include=XMLStatsEm`
2. Press Enter.
3. Monitor the event processing metrics that are reported to ensure that events are processed according to accepted benchmark values.

Sample output after running the XML Stats script

```
<xmlstats created="Tue Oct 23 04:01:25 PDT 2018"
   includes="XMLStatsEm" version="2">
   <itom_event_management>
   <event_processing>
   <processed count="47" max_process_time="10"
      mean_process_time="5" record_time="2018-10-23 11:00:41"/>
   <ready count="7" oldest_ready_event="37" record_time="2018-10-23 11:00:41"/>
   <error count="5" record_time="2018-10-23 11:00:41"/>
   <total_created count="5" record_time="2018-10-23 11:00:41"/>
   </event_processing>
   </itom_event_management>
</xmlstats>
```

The event processing metrics in the report are described in the following table.

```
<table>
<thead>
<tr>
<th>Metric</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event processing statistics</td>
<td></td>
</tr>
<tr>
<td>processed count=n</td>
<td>Number of events processed by the instance.</td>
</tr>
<tr>
<td>max_process_time=n</td>
<td>Longest time taken by the instance to process an event.</td>
</tr>
</tbody>
</table>
```

Note: The default processing cycle is events created in the last hour and updated in the last minute.
### Metric | Description
--- | ---
`mean_process_time=n` | Average time taken by the instance to process the events in this cycle.  
**Event ready statistics**

`ready count=n` | Number of events that are in Ready state and are available to be processed.  
Default processing cycle: events created in the last hour and updated in the last minute.  

`oldest_ready_event=n` | Longest time taken by an event to remain in Ready state. If this metric has a value that is bigger than your benchmark figure, investigate the event further to determine what is impeding the processing.  
Default processing cycle: events created in the last hour.  

**Other processing statistics**

`error count=n` | Number of events for which errors were reported.  
Default processing cycle: events created in the last hour and updated in the last minute.  

`total_created count=n` | Overall number of events that were processed during this event processing cycle.  
Default processing cycle: events created in the last hour.

---

**Enable processing of event process statistics**

Enable the processing of event process statistics.

Role required: `evt_mgmt_admin`

Use the `evt_mgmt.enable_event_processing_stats` property to enable the XMLStatsEm script to run and collect event processing statistics. You can disable the running of the XMLStatsEm script, for example, to alleviate a performance issue.

1. Navigate to **System Properties > All Properties**.
2. Search for the `evt_mgmt.enable_event_processing_stats` property.
3. Either configure the value of an existing `evt_mgmt.enable_event_processing_stats` property to enable the XMLStatsEm script to run or create a new property if it does not exist already:
   - If the property exists, set the property to **Enable** to enable the XMLStatsEm script to collect event processing statistics and click **Update**.
   - If the property does not exist, click **New**, fill in the form, and click **Update**.
### System property fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the property you are creating.</td>
</tr>
<tr>
<td>Description</td>
<td>Narrative that describes the property and its function.</td>
</tr>
<tr>
<td>Choices</td>
<td>Comma-separated values for a choice list. If you need a different choice list label and value, use an equal sign (=) to separate the label from the value. For example, Blue=0000FF, Red=FF0000, Green=00FF00 displays Blue, Red, and Green in the list, and saves the corresponding hex value in the property value field.</td>
</tr>
<tr>
<td>Type</td>
<td>Data type, that is selected from a list (for example, integer, string, or true</td>
</tr>
<tr>
<td>Value</td>
<td>Desired value for the property. When retrieving properties using the gs.getProperty() method, treat the results as strings. For example, a true</td>
</tr>
<tr>
<td>Read roles</td>
<td>Roles that have read access to this property.</td>
</tr>
<tr>
<td>Write roles</td>
<td>Roles that have write access to this property.</td>
</tr>
</tbody>
</table>

### Configure statistics processing period

Set the time period, in seconds, for collecting event processing statistics. For example, you can set a time period twice as long as the default 60 seconds to collect more statistics.

Role required: evt_mgmt_admin

Use the `evt_mgmt.event_processing_stats_period` property to configure the XMLStatsEm script time to collect event processing statistics. Set the property values to run the script over a longer or shorter period. For example, set a shorter period of statistics collection so fewer CPU resources are used.

1. Navigate to System Properties > All Properties.
2. Search for the `evt_mgmt.event_processing_stats_period` property.
3. Either configure the value of an existing `evt_mgmt.enable_event_processing_stats` property to run the XMLStatsEm script over a shorter or longer time period or create a new property if it does not exist already.
   - If the property exists, set the property to the required time period to enable the XMLStatsEm script time to collect event processing statistics and click Update.
   - If the property does not exist, click New, fill in the form, and click Update.
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Narrative that describes the property and its function.</td>
</tr>
<tr>
<td>Choices</td>
<td>Comma-separated values for a choice list. If you need a different choice list label and value, use an equal sign (=) to separate the label from the value. For example, Blue=0000FF, Red=FF0000, Green=00FF00 displays Blue, Red, and Green in the list, and saves the corresponding hex value in the property value field.</td>
</tr>
<tr>
<td>Type</td>
<td>Data type, that is selected from a list (for example, integer, string, or true</td>
</tr>
<tr>
<td>Value</td>
<td>Desired value for the property. When retrieving properties using the gs.getProperty() method, treat the results as strings. For example, a true</td>
</tr>
<tr>
<td>Read roles</td>
<td>Roles that have read access to this property.</td>
</tr>
<tr>
<td>Write roles</td>
<td>Roles that have write access to this property.</td>
</tr>
</tbody>
</table>

### Alert management

As alerts generate, you can view more information about them, acknowledge them, and take action to resolve them. You can also manually create alerts to track issues that did not generate an event or alert.

You can respond to an alert in the following ways:

- Manually remediate the alert.
- Acknowledge an alert that requires attention.
- Create an incident or security incident.
- Create a case.
- Close the alert.
- Resolve any incident that is related to the alert.
- Reopen the alert.

**Note:** Business rules that are written for alert tables (em_alert) must be highly efficient or they may result in performance degradation.

**View alert information**

View a list of all alerts for business services and application services, and then manage individual alerts as necessary.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user
Multiple related events may correlate into a single alert. Event Management only creates alerts when one or more events meet the conditions defined in event rules, alert action rules, and alert configuration settings.

1. Navigate to Event Management > All Alerts.
2. To view or manage an alert, click the alert number.
3. Review the information on the Alert form.
   You can click tabs on the form for further information.
   - To view flapping information, click the Flapping tab.
   - To view alert history, click the History tab.

### Alert form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>If an alert was created as a result of the event, this field contains the unique ID that Event Management generates to identify the alert.</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Node</td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Type</td>
<td>Pre-defined event type, such as high CPU, which is used to identify an event record. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Resource</td>
<td>Node resource that is relevant to the event. For example, Disk C, CPU-1, the name of a process, or service. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>JSON string that represents a configuration item. For example, {&quot;name&quot;:&quot;SAP ORA01&quot;,&quot;type&quot;:&quot;Oracle&quot;}. The CI identifier that generated the event appears in the Additional information field. This field has a maximum length of 1000 digits. Note: Reference pop-ups and click-throughs are hidden by default for read-only fields. For Configuration item and other read only fields, you can optionally change the read-only setting. For more information, see Configure pop-ups on read-only fields. Click to open the alert in dependency view.</td>
</tr>
<tr>
<td>Task</td>
<td>The corresponding task for the alert, such as an incident, change, or problem.</td>
</tr>
<tr>
<td>Metric Name</td>
<td>Unique name that describes which metric data is collected.</td>
</tr>
<tr>
<td>Description</td>
<td>The alert description.</td>
</tr>
<tr>
<td>Message key</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the Source, Node, Type, Resource, and Metric Name field values. This field has a maximum length of 1024 digits.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>The severity of the event. The value for this field is copied from the event unless the event closes the alert, in which case the previous severity is retained for reporting.</td>
</tr>
<tr>
<td></td>
<td>* Critical: Immediate action is required. The resource is either not functional or critical problems are imminent.</td>
</tr>
<tr>
<td></td>
<td>* Major: Major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td></td>
<td>* Minor: Partial, non-critical loss of functionality or performance degradation occurred.</td>
</tr>
<tr>
<td></td>
<td>* Warning: Attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>* Info: An alert is created. The resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>* Clear: No action is required. An alert is not created from this event. Existing alerts are closed.</td>
</tr>
<tr>
<td>State</td>
<td>The state of the alert.</td>
</tr>
<tr>
<td></td>
<td>* Open: The alert requires user action.</td>
</tr>
<tr>
<td></td>
<td>* Reopen: The previously closed alert requires additional user action.</td>
</tr>
<tr>
<td></td>
<td>* Flapping: After the alert has been closed, it receives a high frequency of identical events from the same source that causes many alert reopenings. User action is required.</td>
</tr>
<tr>
<td></td>
<td>* Closed: The alert is closed and no further user action is required.</td>
</tr>
<tr>
<td>Acknowledged</td>
<td>Select to show that a user has acknowledged the alert.</td>
</tr>
<tr>
<td>Maintenance</td>
<td>A check box that shows whether the resource is affected by the alert is in maintenance.</td>
</tr>
<tr>
<td>Updated</td>
<td>The most recent time that the alert information was updated.</td>
</tr>
<tr>
<td>Parent</td>
<td>If this field is blank, you can specify an alert number. The current alert becomes a secondary alert to this parent alert.</td>
</tr>
<tr>
<td></td>
<td>If an alert number is displayed, then the current alert is already a secondary alert to this current alert, which is the parent alert.</td>
</tr>
<tr>
<td>Feedback</td>
<td>This field appears in alert forms only if the alert being modified is a parent alert. Your response provides feedback about the accuracy and usefulness of the group. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>* None - No value was selected.</td>
</tr>
<tr>
<td></td>
<td>* No Feedback - There is no feedback for this alert.</td>
</tr>
<tr>
<td></td>
<td>* Yes - The alert group is accurate and useful.</td>
</tr>
<tr>
<td></td>
<td>* No - The alert group is not accurate and not useful.</td>
</tr>
<tr>
<td>Knowledge article</td>
<td>The knowledge article associated with the alert, if any.</td>
</tr>
</tbody>
</table>
## Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Event Count</td>
<td>The counter is increased every time an event is bound to the alert. The count value is kept even after the actual event has been purged. Primary alerts (virtual alerts) are updated based on their secondary alerts. The actual number of events that affected the alert is displayed.</td>
</tr>
<tr>
<td>Source instance</td>
<td>The name of the machine or software that generated the event. For example, SolarWinds on 10.22.33.44.</td>
</tr>
<tr>
<td>User name and role</td>
<td>The user and role of the person who made the most recent alert updates.</td>
</tr>
<tr>
<td>Acknowledged</td>
<td>The <strong>Acknowledged</strong> check box value after the most alert recent update.</td>
</tr>
<tr>
<td></td>
<td>• <strong>True</strong>: The <strong>Acknowledged</strong> is selected.</td>
</tr>
<tr>
<td></td>
<td>• <strong>False</strong>: The <strong>Acknowledged</strong> check box is cleared.</td>
</tr>
<tr>
<td>Correlated Alerts section</td>
<td>The secondary alerts that are correlated with this alert, where this alert is the primary alert. For more information, see Alert correlation rules.</td>
</tr>
<tr>
<td>Primary Alerts section</td>
<td>The primary alert that is correlated with this alert, where this alert is a secondary alert. For more information, see Alert correlation rules.</td>
</tr>
<tr>
<td>Impacted Services tab</td>
<td>The impacted services that are related to this alert are listed. For more information, see View the impact tree.</td>
</tr>
<tr>
<td>Flap count</td>
<td>The number of times the alert has flapped—that is, has fluctuated between a closed and a non-closed state—within the flap interval since the start time in the Flap start window.</td>
</tr>
<tr>
<td>Flap start window</td>
<td>The initial start time to measure the flapping occurrences.</td>
</tr>
<tr>
<td>Flap last update time</td>
<td>The last time flapping occurred. This time is the ServiceNow processing time, not the source system time.</td>
</tr>
<tr>
<td>Flap last state</td>
<td>The state before the alert entered the flapping state.</td>
</tr>
<tr>
<td>Initial event generation time</td>
<td>The time when the event that generated the alert first occurred. This time is the ServiceNow processing time, not the source system time.</td>
</tr>
</tbody>
</table>

**Note:** Do not create an Alert Group based on Event Count. Where an Alert Group is based on Event Count, the impact of the group does not calculate properly because the Event Count field is not copied to the em_alert_history table for impact calculation.

**Additional tabs.**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impacted Services</td>
<td>The impacted services that are related to this alert are listed. For more information, see View the impact tree.</td>
</tr>
<tr>
<td>Flap count</td>
<td>The number of times the alert has flapped—that is, has fluctuated between a closed and a non-closed state—within the flap interval since the start time in the Flap start window.</td>
</tr>
<tr>
<td>Flap start window</td>
<td>The initial start time to measure the flapping occurrences.</td>
</tr>
<tr>
<td>Flap last update time</td>
<td>The last time flapping occurred. This time is the ServiceNow processing time, not the source system time.</td>
</tr>
<tr>
<td>Flap last state</td>
<td>The state before the alert entered the flapping state.</td>
</tr>
<tr>
<td>Initial event generation time</td>
<td>The time when the event that generated the alert first occurred. This time is the ServiceNow processing time, not the source system time.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last event generation time</td>
<td>The last time the event that is linked to the alert occurred. This time is the ServiceNow processing time, not the source system time.</td>
</tr>
<tr>
<td>Created</td>
<td>The alert creation time.</td>
</tr>
<tr>
<td>Work notes</td>
<td>The additional notes about the alert.</td>
</tr>
<tr>
<td>Activities tab</td>
<td>A record of the work that was performed. The text has a date and time stamp.</td>
</tr>
<tr>
<td>Activity</td>
<td>The additional notes about the alert.</td>
</tr>
<tr>
<td>More Information tab</td>
<td>Displays the computation of the alert priority score.</td>
</tr>
<tr>
<td>Priority Breakdown</td>
<td>A JSON string that gives more information about the event. The JSON data is supported for String values only, other value types are not supported. You must convert numbers to String values by enclosing them in double quotes. For example, this value is not supported: <code>{&quot;CPU&quot;:100}</code> while this value is supported: <code>{&quot;CPU&quot;:&quot;100&quot;}</code>. Another example of a valid JSON string is: <code>{&quot;evtComponent&quot;:&quot;Microsoft-Windows-UpdateClient&quot;,&quot;evtMessage&quot;:&quot;Installation Failure: Windows failed. Error 0x80070490&quot;}; This information can be used for third-party integration or other post-alert processing. Values in the **Additional information** field of an Event that are not in JSON key/value format are normalized to JSON key/value format when the event is processed. For example, assume that the following plain text is in the **Additional information** field &quot;Connection instance is successful&quot;. When the event is processed, all this plain text becomes one JSON string and might not be useful within an alert. In the resultant alert, this string is in the **Additional information** field in JSON key/value format, containing the data: </code>{&quot;additional_content&quot;:&quot;Connection instance is successful&quot;}`.</td>
</tr>
</tbody>
</table>

For more information about the Calculate Related Tasks command and the alert insight tabs, see [View alert insight information](#).

You can respond to the alert in the following ways:

#### Alert response options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submit</td>
<td>Save the modifications that were made to the form and return to the <strong>Alerts</strong> list.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Acknowledge the alert.</td>
<td>Click <strong>Acknowledge</strong>. If the alert is reopened, this button reappears so you can reacknowledge the alert.</td>
</tr>
<tr>
<td>Create an incident.</td>
<td>Click <strong>Create incident</strong>. For more information, see <a href="#">Create an incident or security incident from an alert</a>.</td>
</tr>
<tr>
<td>If Security Incident Response is activated, create a security incident response.</td>
<td>Click <strong>Create Security Incident</strong>.</td>
</tr>
<tr>
<td>Designate that the alert is in maintenance.</td>
<td>Select the Maintenance check box. For more information, see <a href="#">View all alerts by the maintenance status</a>.</td>
</tr>
<tr>
<td>Close the alert.</td>
<td>Click <strong>Close</strong>. For more information, see <a href="#">Close an Event Management alert</a>.</td>
</tr>
</tbody>
</table>

Alert execution information
Alert execution information provides a reference to the actions that have been performed concerning the alert. Among the information presented is which alert management rules ran on the alert, incidents that were opened, and which remediation workflows ran.

**Alert Executions**
In an alert form, the Alert Execution tab shows key related records for the alert. For example, it shows which actions were performed and which alert management rules were run. If an incident was opened, the related tasks are shown.

### Alert Executions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert</td>
<td>Reference to the alert that ran under this rule.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Action name</td>
<td>Indicates how the alert was resolved.</td>
</tr>
<tr>
<td>Link to execution</td>
<td>Link action that was taken to resolve the alert.</td>
</tr>
<tr>
<td>Related task</td>
<td>Link to the incident that was created when the alert was resolved.</td>
</tr>
<tr>
<td>Log</td>
<td>Link to the log that details the actions taken to resolve the alert.</td>
</tr>
<tr>
<td>Automatic run</td>
<td>Indicates if the rule was automatic (true) or manual (false).</td>
</tr>
</tbody>
</table>

**Action name**

This indicates what occurred, for example, if an incident was created.

**Log**

The log reveals which alert management rule initiated the action. It shows, for example, automatic execution 1 out 3, or manual execution, 2 out of 4.

**Alert insight information**

Alert insight aids faster alert triage, enabling a quicker way to find a solution and expose the probable root cause of the selected alert.

**Use alert insight to find a solution and help find the root cause of the alert**

Results from data analysis provide further information about the alert, beyond that of the alert form. This aids faster alert triage, enabling a quicker way to find a solution and expose the probable root cause of the alert.

Information shown is from repeated alerts, similar alerts, incidents, problems, and change requests on the same CI, similar CIs, and other CIs that are related by CMDB relations and business service (BS) relations, or grouping. For more information about the alert insight tabs in the alert form, see [View alert insight information](#).

**View alert insight information**

In the alert form, in the alert insight tabs, view calculated insight information on the alert and the CI, based on history and analysis of CMDB and ITSM data. Use alert insight information to show how to resolve the alert and help find what the root cause is.

**Role required: evt_mgmt_user**

The alert insight tabs contain insights on the alert and the related CIs based on history and analysis of CMDB and ITSM data. The alert insight information can help to find a solution for the alert and expose the probable root cause of the alert.

Domain separation is supported. For more information, see [Domain separation and Event Management](#).

1. Navigate to **Event Management > All Alerts**.
2. Click the required alert number.
3. Review the alert insight information on the alert form in the tabs that appear in the red frame in the following image:
## Alert insight tabs in the alert form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>When calculated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeated Alerts</td>
<td>List of alerts that have the same message key as that of the selected alert CI and that are in closed state. Use this insight information to find a solution for the alert. Timeframe: Alerts that are up to 30 days old are retrieved.</td>
<td>When the alert form is refreshed.</td>
</tr>
<tr>
<td>Similar Alerts</td>
<td>List of similar alerts that have the same CI type as that of the selected alert CI, as well as the same source, resource, metric name, and so on. Use this insight information to find a solution for the alert and to see similar alerts that were resolved. Timeframe: Alerts that are up to 30 days old are retrieved. For a list of other configurable similar fields, see the description of the <code>evt_mgmt.alert_insight_alert_same_as_filter</code> property in the <a href="#">Alert insight properties</a> topic.</td>
<td>When the alert form is refreshed.</td>
</tr>
<tr>
<td>CI Incidents</td>
<td>List of incidents that have the same CI as that of the selected alert CI. Timeframe: Incidents that were created up to 7 days before the alert was created are retrieved.</td>
<td>When the alert form is refreshed.</td>
</tr>
<tr>
<td>CI Change Requests</td>
<td>List of change requests that have the same CI as that of the selected alert CI. Change requests are only retrieved for these states: implement, closed, and review. Use this insight information to find the root cause of the alert. Timeframe: Incidents that were created up to 7 days before the alert was created are retrieved.</td>
<td>When the alert form is refreshed.</td>
</tr>
<tr>
<td>CI Problems</td>
<td>List of problems that have the same CI as that of the selected alert CI. Use this insight information to find the root cause of the alert. Timeframe: Incidents that were created up to 7 days before the alert was created are retrieved.</td>
<td>When the alert form is refreshed.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td>When calculated</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Related Incidents</td>
<td>List of incidents on CIs that have some level of relationship with the selected alert CI. Use this insight information to find the root cause of the alert.</td>
<td>Calculated on demand by clicking Calculate Related Tasks.</td>
</tr>
<tr>
<td></td>
<td>Timeframe: Incidents that are up to 7 days old from the creation time of the alert are retrieved.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>When the Related Incidents tab is selected:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Use the value in the <strong>Score</strong> column to see which task might be more relevant as the root cause for the alert. For more information, see <a href="#">Score column</a>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Use the value in the <strong>Affiliation Type</strong> column to see the CI relationships. For more information, see <a href="#">Affiliation Type column</a>.</td>
<td></td>
</tr>
<tr>
<td>Related Change Requests</td>
<td>List of change requests on CIs that have a level of relationship with the selected alert CI. Use the insight information to find the root cause of the alert.</td>
<td>Calculated on demand by clicking Calculate Related Tasks.</td>
</tr>
<tr>
<td></td>
<td>Timeframe: Change requests that are up to 7 days old from the creation time of the alert are retrieved.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>When the Related Change Requests tab is selected:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Use the value in the <strong>Score</strong> column to see which task might be more relevant as the root cause for the alert. For more information, see <a href="#">Score column</a>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Use the value in the <strong>Affiliation Type</strong> column to see the CI relationships. For more information, see <a href="#">Affiliation Type column</a>.</td>
<td></td>
</tr>
<tr>
<td>Related Problems</td>
<td>List of problems on CIs that have a level of relationship with the selected alert CI. Use this insight information to find the root cause of the alert.</td>
<td>Calculated on demand by clicking Calculate Related Tasks.</td>
</tr>
<tr>
<td></td>
<td>Timeframe: Problems that are up to 7 days old from the creation time of the alert are retrieved.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>When the Related Problems tab is selected:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Use the value in the <strong>Score</strong> column to see which task might be more relevant as the root cause for the alert. For more information, see <a href="#">Score column</a>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>· Use the value in the <strong>Affiliation Type</strong> column to see the CI relationships. For more information, see <a href="#">Affiliation Type column</a>.</td>
<td></td>
</tr>
</tbody>
</table>

**Related CIs**

For further information about the various relations in the **Related Incidents**, **Related Change Requests**, and **Related Problems** columns, see the following table.
CI affiliation type relationship

<table>
<thead>
<tr>
<th>Value</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CMDB dependent relationship rules</strong></td>
<td>• Regular CMDB relation rules</td>
</tr>
<tr>
<td></td>
<td>• Metadata rules (these rules help identify dependent CIs):</td>
</tr>
<tr>
<td></td>
<td>• Containment rules—these rules represent configuration hierarchy of CIs, describing which CI contains which other CIs</td>
</tr>
<tr>
<td></td>
<td>• Hosting rules—these rules represent the placement of CIs in a business definition and describes what the CIs run on</td>
</tr>
<tr>
<td></td>
<td>• Suggested rules</td>
</tr>
<tr>
<td>Business service</td>
<td>CIs that are in the same business service.</td>
</tr>
<tr>
<td>Alert group</td>
<td>CIs of alerts within the same group as that of the selected alert.</td>
</tr>
</tbody>
</table>

The **Score** column in the **Related Incidents**, **Related Change Requests**, and **Related Problems** columns indicate which task might be more relevant as the root cause for the alert. The higher the value of the score, the more relevant that task is as the root cause of the alert. The value is the level of relationship in CMDB or business service, plus its value if it exists in the same alert group.

The score value is calculated as follows:

- The CI found in level 1 has a score of 3.
- The CI found in level 2 has a score of 2.
- The CI found in level 3 has a score of 1.
- The CI that is found in the alert group (the alert group that contains the alert CI) has a score of 2.

For example, if a CI at level 3 is found in the alert group and it has a relationship with the alert CI, the score value is the alert group score plus the level value of the CI. In this example, the score is 1 plus 2 = 3 (the task level plus the alert group score).

The **Affiliation Type** column in the **Related Incidents**, **Related Change Requests**, and **Related Problems** columns show the type of relationship that the alert CI and the CI have. To see affiliation type relationships, navigate to **Configuration > Identification/Reconciliation > Metadata Editor**.

The parent-child relationship between configuration items is considered:

- Containment rules represent configuration hierarchy of CIs, describing which CI contains which other CIs.
- Hosting rules represent the placement of CIs in a business definition and describes what the CIs run on.

For further information, see **CMDB dependent relationship rules**.

Use properties to configure alert insight. For more information, see **Alert insight properties**.

**Alert insight properties**

Use these properties to configure alert insight.

- The role required: evt_mgmt_admin
- The following alert_insight properties are under sys_properties.
<table>
<thead>
<tr>
<th>Property</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Frame</td>
<td></td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_alert_history_min</td>
<td>Set the time frame (in minutes) to retrieve repeated and similar alert data. Default 43200 (30 days)</td>
</tr>
<tr>
<td></td>
<td>Note: Alerts are retrieved regardless of their state (open / reopen / flapping / closed).</td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_closed_alert_window</td>
<td>Set the time frame (in minutes) to retrieve alerts that were already closed. It is the time after the alert last updated date. Default: 4320 (3 days)</td>
</tr>
<tr>
<td>Similarity</td>
<td></td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_alert_same_as_filter</td>
<td>This property is a comma-separated string that defines which of the alert fields is used to consider alerts to be similar. Default: source, type, resource, metric_name</td>
</tr>
<tr>
<td>Related CIs</td>
<td></td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_related_cis_topology_levels</td>
<td>The relationship types are:</td>
</tr>
<tr>
<td></td>
<td>· CMDB based (metadata rules and suggested relations)</td>
</tr>
<tr>
<td></td>
<td>· Within business services</td>
</tr>
<tr>
<td></td>
<td>· Within alert groups</td>
</tr>
<tr>
<td></td>
<td>For ‘Within business service’ relationship type, this property sets the depth or the maximum level of relationship of retrieved CIs. Default: 3</td>
</tr>
<tr>
<td>Score</td>
<td></td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_group_mapping</td>
<td>This property sets the score for within alert group relations. Default: 2</td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_level_1_mapping</td>
<td>This property sets the score for level 1 relationship. Default: 3</td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_level_2_mapping</td>
<td>This property sets the score for level 2 relationship. Default: 2</td>
</tr>
<tr>
<td>evt_mgmt.alert_insight_level_3_mapping</td>
<td>This property sets the score for level 3 relationship. Default: 1</td>
</tr>
<tr>
<td>Maximum related tasks</td>
<td>Maximum related tasks to retrieve for alert insight. Default: 10</td>
</tr>
</tbody>
</table>

**Metadata rules consideration**

The parent-child relationship of CIs is considered. Dependent relationship rules consist of hosting and containment rules, each type modeling the data from a different perspective of the CI.

To manage dependent relationship rules:

- To access rules at the class level and use the CI Class Manager. Navigate to **Configuration > CI Class Manager**.
To access grouped rules, use the Metadata Editor. Navigate to Configuration > Identification/Reconciliation > Metadata Editor.

Containment rules represent configuration hierarchy of CIs, describing which CI contains which other CIs.

Hosting rules represent placement of CIs in a business definition, describing what CIs run on.

Modify the alert insight properties to configure the way alert information and analysis appears in the Alert Insight pane.

**Related CIs configuration**

The following properties control which CMDB relationships to consider for related CIs. The CMDB relationships include regular CMDB relation rules, metadata rules (containment rules and hosting rules), and suggested relations.

<table>
<thead>
<tr>
<th>Property</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>evt_mgmt.related_cis_get_all_relation_types</td>
<td>Get all relation types, not including metadata rules. Default: false</td>
</tr>
<tr>
<td>evt_mgmt.related_cis_use_containment_rules</td>
<td>Use metadata containment rules. Default: true</td>
</tr>
<tr>
<td>evt_mgmt.related_cis_use_hosting_rules</td>
<td>Use metadata hosting rules. Default: true</td>
</tr>
<tr>
<td>evt_mgmt.related_cis_use_suggested_relations_rules</td>
<td>Use suggested relations rules. Default: false</td>
</tr>
<tr>
<td>evt_mgmt.related_cis_validate_relation_rules</td>
<td>This property controls whether to validate relation of CI according to metadata rules. Default: true</td>
</tr>
</tbody>
</table>

**Score**

Scores are configured per relation type or depth. Scores are accumulated. The higher the score, the more relevant is the related CI to the current CI.

**Example:** For a CI that was found at level 2 in the same business service of the current CI, the score is 2. The same CI is in the same alert group, so there is an extra score of 3. The accumulated score is therefore $2 + 3 = 5$.

**Affiliation type**

The Affiliation Type column in the Related Incidents, Related Change Requests, and Related Problems tabs shows the type of relationship that the CI of the selected alert and the related CI have.

To see affiliation type relationships, navigate to Configuration > Identification/Reconciliation > Metadata Editor.

The parent-child relationship between configuration items is considered.

**Alert process flow**

Alert management rules and alert correlation rules run on the Alert (em_alert) table.
Alert process flow

Actions are taken on alerts according to this process flow:

- Match the alert management rule to an alert.
  - If the source of the event matches the source specified in an existing rule, then a rule is matched. You can define any kind of condition, on the source of the alert or any other field, and combination of fields.
  - If multiple alert management rules are defined for the same type of alert, use the rule **Order** to determine the order of rule application.
- Match the alert correlation rule to an alert.

![Alert process flow diagram](image-url)
Task/Incident fields that are populated from the alert by default

- `alert.description` is copied to `incident.short_description` and `incident.description`
- `alert.cmdb_ci` is copied to `incident.cmdb_ci`
- `alert.severity` is transformed into `incident.urgency`

For automatically opened incidents:
- `Alert value` is copied to `incident.contact_type`
- `sys_user value` is copied to `incident.caller_id`

Business Rules/Jobs that perform the alert processing actions

<table>
<thead>
<tr>
<th>Action</th>
<th>Name of Business Rule/Job</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply alert template</td>
<td>Apply overwrite rule and validate</td>
<td>Business rule</td>
</tr>
<tr>
<td>Open incident</td>
<td>Event Management - create/resolved incidents by alerts</td>
<td>Scheduled job</td>
</tr>
<tr>
<td>Fill KB</td>
<td>Event Management - create/resolved incidents by alerts</td>
<td>Scheduled job</td>
</tr>
<tr>
<td>Acknowledge</td>
<td>Event Management - create/resolved incidents by alerts</td>
<td>Scheduled job</td>
</tr>
<tr>
<td>Remediation</td>
<td>Run automatic remediation actions</td>
<td>Business rule</td>
</tr>
<tr>
<td>Reopen incident</td>
<td>Reopen associated closed incident</td>
<td>Business rule</td>
</tr>
<tr>
<td>Close Alert by closed incident</td>
<td>Event Management - create/resolved incidents by alerts</td>
<td>Scheduled job</td>
</tr>
<tr>
<td>Close incident by closed Alert</td>
<td>Close associated incident</td>
<td>Business rule</td>
</tr>
</tbody>
</table>

Invoke domain separation through subflows

Event Management supports domain separation in both its out-of-the-box and customized subflows. After customizing a subflow, add it to the Domain Separation Subflows White List table (`em_ds_subflow_white_list`).

Only out-of-the-box subflows in the Domain Separation Subflows White List table (`em_ds_subflow_white_list`) are verified for domain separation. When customizing subflows, an admin user must verify that the subflows support domain separation before adding them to the Domain Separation Subflows White List table (`em_ds_subflow_white_list`).

View alerts in the flapping state
You can view alerts that are specifically in the flapping state.

Before starting this procedure, ask your administrator to configure alert flapping properties.

Role required: `evt_mgmt_admin`, `evt_mgmt_operator`, or `evt_mgmt_user`

Flapping condition that occurs when the event source continues to generate events even after its associated alert has been closed. Flapping causes the status of the resource to repeatedly fluctuate between an Info Severity and another severity that requires attention, for example, Critical.
The frequency of events from an identical source within a given time interval determines whether an alert is in a flapping state or a new issue has occurred. Based on the `evt_mgmt.flap_frequency` and `evt_mgmt.flap_interval` property values:

- If the same issue is recurring, Event Management associates the new event with the existing alert and the state of the alert is set to **Flapping**.
- If the issue occurs after the time interval expires, Event Management creates an alert.

For example, you can respond to an alert by rebooting a problematic server. After no events are generated for several minutes, it is assumed that the issue is fixed and the alert is closed. If the reboot did not actually fix the issue, this server can generate more events later. Then additional alerts are generated for the same issue.

1. Navigate to **Event Management > All Alerts**.
2. Click the number of an alert that is in the **Flapping** state.
3. On the alert, click the **Flapping** tab.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flapping tab</td>
<td></td>
</tr>
<tr>
<td>Flap count</td>
<td>The number of times the alert has flapped—that is, has fluctuated between a closed and a non-closed state—within the flap interval since the start time in the Flap start window.</td>
</tr>
<tr>
<td>Flap start window</td>
<td>The initial start time to measure the flapping occurrences.</td>
</tr>
<tr>
<td>Flap last update time</td>
<td>The last time flapping occurred. This time is the ServiceNow processing time, not the source system time.</td>
</tr>
<tr>
<td>Flap last state</td>
<td>The state before the alert entered the flapping state.</td>
</tr>
</tbody>
</table>

4. If the **Parent** field is empty, address this alert as a new issue.

**View all alerts by the maintenance status**

The **Maintenance** status indicates that the CI is under maintenance. For example, there is a software upgrade, and the issues can result from that activity, therefore all maintenance alerts are discarded.

Role required: `evt_mgmt_admin`, `evt_mgmt_operator`, or `evt_mgmt_user`

The alert maintenance status can change due to any of these reasons:

- A user selected the **Maintenance** check box on the alert, or selected the alert from the list view, and then clicked the **Maintenance** UI action.
- A new alert that had been bound to a CI whose status is in maintenance status on the CMDB table.
- If all previous events associated with the alert are in maintenance, the alert **Maintenance** field can automatically be set to **true** showing maintenance is in progress. However, if an alert **Maintenance** field is **false** but the related CI status is changed to **In Maintenance** after the alert was created, any subsequent events do not put the alert into maintenance. For example, if an alert is generated for a router and later the router is taken down for maintenance, it is still assumed that there was a problem before maintenance started and the issue must not be ignored.
Note: If a maintenance flag was set on an alert in Geneva, after upgrading to Helsinki it affects the impact as a regular alert. However, it is seen in the alert console only when a new event arrives.

1. Navigate to Event Management > All alerts.
2. Review the Maintenance column for each alert.
   A value of true indicates that the CI is under maintenance.

Note: If the Maintenance field is not visible on the Alert form, personalize the list to add the field. For more information, see Personal lists.

Create maintenance rules
Use maintenance rules to mark CIs in maintenance status. When in maintenance status, these CIs are excluded from impact calculation.

Role required: evt_mgmt_admin

You can define rules to mark CIs that match the specified criteria as being in maintenance status. The marked CIs populate the Impact Maintenance CIs (em_impact_maint_ci) table.

Note: When running maintenance rules, the cmdb_ci status of matching CIs is not changed. However, matching CIs are flagged in the em_impact_maint_ci table by these rules and this status is considered for impact and alert calculations.

The field flagged by this rule as being in maintenance status is the CMDB status field.

Note: To activate a maintenance rule, select the Active field in the maintenance rule. To deactivate a maintenance rule, clear the Active field.

The maintenance rules provided with the base instance are:
### Default maintenance rule

**CI in Change Window**

Where the CI has an active change window, the matching CIs are marked as being in maintenance status.

The rule runs a query against the change request (change_request) table to determine whether the rule is applied. All these conditions in the change_request table must be met:

- **State** is one of these options: Scheduled, Implement, Work in Progress, or Open/New (state in (-2, -1, 1, 2)).
- **Approval** is Approved (approval = 'approved').
- The change request window is active, that is, the current time is between Planned start date and Planned end date, or the current time is between Actual start date and Actual end date.
- The change request record is not an on-hold record (on_hold='false').

**Note:** All these conditions must be present for the CI to be placed in maintenance status by this rule. For example, if the State of the change request approval status is Change is waiting for approval, then the change is not added to the em_impact_maint_ci table.

### Maintenance status of CI

CIs whose CMDB status field is **In Maintenance** are flagged by this rule as being in maintenance status.

By default, retired CIs are not included in CIs that are **In Maintenance**. To include retired CIs:

1. Navigate to **All Properties**.
2. Select **New**.
3. Configure the **evt_mgmt.maintenance_rule_include_retired** property with **Value = true**.

**Note:** Including thousands of retired CIs may decrease system performance.

1. Navigate to **Event Management > Rules > Maintenance Rules**.
2. Click **New**.
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Column heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The maintenance rule name.</td>
</tr>
<tr>
<td>Active</td>
<td>Select to activate the maintenance rule.</td>
</tr>
<tr>
<td>Advanced</td>
<td>Select to enable the optional script section to display.</td>
</tr>
<tr>
<td>Column heading</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>Information that describes this maintenance rule.</td>
</tr>
<tr>
<td>Flag CIs that run on this host</td>
<td>Select to flag all applications running on the recognized host as being in maintenance.</td>
</tr>
<tr>
<td>Table</td>
<td>Select the table that contains the CI that you require.</td>
</tr>
<tr>
<td>Filter</td>
<td>Specify how to select the data.</td>
</tr>
<tr>
<td>CI field name</td>
<td>Select the CI from the list. The list is populated according to your selection in the Table field. If a CMDB table, or a table derived from CMDB, was selected, specify <code>sys_id</code> for the CI field name. Otherwise, specify the required CI field that you want to use. See the examples.</td>
</tr>
</tbody>
</table>

4. **Click Submit.**

Example of a maintenance rule using a CMDB table.

Assume that a company defines a CI as being in maintenance when the Operational Status of the CI is either **Repair in Progress** or **DR Standby**.
Example of a maintenance rule using a table other than CMDB.

Assume that a company uses Incident records to track maintenance. Any maintenance request is translated to an incident when it has a description that starts with “Performing maintenance on CI”. As long as the status of such an incident is open, this status indicates that the maintenance is in progress. For the CI field name field, specify a CI name from the table that was chosen, in this case Incident.
Example of a maintenance rule that uses the advanced script feature.

In the Maintenance Rule page, select the **Advanced** option. Customize the provided example script:

```javascript
var gr = new GlideRecord('cmdb_ci');
gr.addQuery('name', 'your_name_here');
gr.query();
while (gr.next()){
  result.push(gr.sys_id + '');}
```

The return value for this example script is a text string that represents an array of CI IDs, for example, `['sys_id1', 'sys_id2', 'sys_id3']`.

You can use this script as an example to prepare your own customized script.
Place an alert into maintenance
You can manually place any alert into maintenance to hide it from the alert console and the Alerts tab.

Role required: evt_mgmt_admin, evt_mgmt_operator

Placing the alert in maintenance does not place the CI into maintenance. The Status field on the CI record remains the same.

1. Navigate to Event Management > All Alerts.
2. Perform one of the following actions:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>To put multiple alerts into maintenance</td>
<td>Select the check boxes next to each alert, and then click the Maintenance UI action at the top of the list.</td>
</tr>
<tr>
<td>To put one alert into maintenance</td>
<td>Open the alert, and either select the Maintenance check box and click Update, or click the Maintenance UI action at the top of the list.</td>
</tr>
</tbody>
</table>

Collaborate from within an alert
You can collaborate with colleagues and write work notes while working in an alert.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

You can select to work either in a compact view of Connect that overlays the standard user interface or in a full-screen workspace. The conversations are not copied into the alert.

1. Navigate to Event Management > All Alerts.
2. Click the required alert.
3. In the alert screen, click Follow.
The label of the button changes to **Following**, showing the current connection to collaboration status.

4. To view the Connect sidebar in a full window, click the down arrow in the **Follow** button and select **Open Connect Full**.

   You can write text in the **Worknote** field. This text is not copied into the alert.

5. To collaborate with colleagues or with support, click the **Add User** icon.

6. To conclude the Connect collaboration session, click **Following**.

**Manually create an alert**

An alert can be automatically generated or you can manually create an alert. After creating the alert, an event is also created in the Event (em_event) table. Alerts that are manually created are useful for testing purposes.

**Role required: evt_mgmt_admin or evt_mgmt_operator**

1. Navigate to **Event Management > All Alerts**
2. Click **New**.
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>If an alert was created as a result of the event, this field contains the unique ID that Event Management generates to identify the alert.</td>
</tr>
<tr>
<td>Source</td>
<td>Event monitoring software that generated the event, such as SolarWinds or SCOM. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Node</td>
<td>Node name, fully qualified domain name (FQDN), IP address, or MAC address that is associated with the event, such as IBM-ASSET. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Type</td>
<td>Pre-defined event type, such as high CPU, which is used to identify an event record. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Resource</td>
<td>Node resource that is relevant to the event. For example, Disk C, CPU-1, the name of a process, or service. This field has a maximum length of 100 digits.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>JSON string that represents a configuration item. For example, ({&quot;name&quot;:&quot;SAP ORA01&quot;,&quot;type&quot;:&quot;Oracle&quot;}). The CI identifier that generated the event appears in the Additional information field. This field has a maximum length of 1000 digits. <strong>Note:</strong> Reference pop-ups and click-throughs are hidden by default for read-only fields. For <strong>Configuration item</strong> and other read only fields, you can optionally change the read-only setting. For more information, see <a href="#">Configure pop-ups on read-only fields</a>.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Task</td>
<td>Event Management SLA task for CIs and business services. This is an activity that is associated with this alert. For example, a change request to roll back the database instance to an earlier version.</td>
</tr>
<tr>
<td>Description</td>
<td>The alert description.</td>
</tr>
<tr>
<td>Metric Name</td>
<td>Unique name that describes which metrics are collected and for which this alert has been created.</td>
</tr>
<tr>
<td>Message key</td>
<td>Unique event identifier to identify multiple events that relate to the same alert. If this value is empty, it is generated from the Source, Node, Type, Resource, and Metric Name field values. This field has a maximum length of 1024 digits.</td>
</tr>
<tr>
<td>Severity</td>
<td>The severity of the event. The value for this field is copied from the event unless the event closes the alert, in which case the previous severity is retained for reporting.</td>
</tr>
<tr>
<td></td>
<td>- Critical: Immediate action is required. The resource is either not functional or critical problems are imminent.</td>
</tr>
<tr>
<td></td>
<td>- Major: Major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td></td>
<td>- Minor: Partial, non-critical loss of functionality or performance degradation occurred.</td>
</tr>
<tr>
<td></td>
<td>- Warning: Attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>- Info: An alert is created. The resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>- Clear: No action is required. An alert is not created from this event. Existing alerts are closed.</td>
</tr>
<tr>
<td>State</td>
<td>The state of the alert.</td>
</tr>
<tr>
<td></td>
<td>- Open: The alert requires user action.</td>
</tr>
<tr>
<td></td>
<td>- Reopen: The previously closed alert requires additional user action.</td>
</tr>
<tr>
<td></td>
<td>- Flapping: After the alert has been closed, it receives a high frequency of identical events from the same source that causes many alert reopenings. User action is required.</td>
</tr>
<tr>
<td></td>
<td>- Closed: The alert is closed and no further user action is required.</td>
</tr>
<tr>
<td>Acknowledged</td>
<td>Select to show that a user has acknowledged the alert.</td>
</tr>
<tr>
<td>Maintenance</td>
<td>A check box that shows whether the resource is affected by the alert in maintenance.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Updated</td>
<td>The most recent time that the alert information was updated.</td>
</tr>
<tr>
<td>Parent</td>
<td>The alert number of the parent alert that this alert is secondary to.</td>
</tr>
<tr>
<td></td>
<td>This field appears only when this alert is secondary to another alert.</td>
</tr>
<tr>
<td>Knowledge Article</td>
<td>The knowledge article associated with the alert, if any.</td>
</tr>
<tr>
<td>Impacted Services tab</td>
<td></td>
</tr>
<tr>
<td>Impacted Services</td>
<td>System generated content.</td>
</tr>
<tr>
<td>Flapping tab</td>
<td></td>
</tr>
<tr>
<td>Flap count</td>
<td>The number of times the alert has flapped—that is, has fluctuated between</td>
</tr>
<tr>
<td></td>
<td>a closed and a non-closed state—within the flap interval since the start</td>
</tr>
<tr>
<td></td>
<td>time in the Flap start window.</td>
</tr>
<tr>
<td>Flap start window</td>
<td>The initial start time to measure the flapping occurrences.</td>
</tr>
<tr>
<td>Flap last update time</td>
<td>The last time flapping occurred. This time is the ServiceNow processing</td>
</tr>
<tr>
<td></td>
<td>time, not the source system time.</td>
</tr>
<tr>
<td>Flap last state</td>
<td>The state before the alert entered the flapping state.</td>
</tr>
<tr>
<td>History tab</td>
<td></td>
</tr>
<tr>
<td>Initial event</td>
<td>The time when the event that generated the alert first occurred. This time</td>
</tr>
<tr>
<td>generation time</td>
<td>is the ServiceNow processing time, not the source system time.</td>
</tr>
<tr>
<td>Last event generation</td>
<td>The last time the event that is linked to the alert occurred. This time is</td>
</tr>
<tr>
<td>time</td>
<td>the ServiceNow processing time, not the source system time.</td>
</tr>
<tr>
<td>Created</td>
<td>The alert creation time.</td>
</tr>
<tr>
<td>Work notes</td>
<td>The additional notes about the alert.</td>
</tr>
<tr>
<td>Activities tab</td>
<td></td>
</tr>
<tr>
<td>Activity</td>
<td>System generated content.</td>
</tr>
<tr>
<td>Additional tab</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Additional information</td>
<td>A JSON string that gives more information about the event. The JSON data is supported for String values only, other value types are not supported. You must convert numbers to String values by enclosing them in double quotes. For example, this value is not supported: {&quot;CPU&quot;:100 } while this value is supported: {&quot;CPU&quot;:&quot;100&quot;}. Another example of a valid JSON string is: {&quot;evtComponent&quot;:&quot;Microsoft-Windows-WindowsUpdateClient&quot;,&quot;evtMessage&quot;:&quot;Installation Failure: Windows failed. Error 0x80070490&quot;}. This information can be used for third-party integration or other post-alert processing. Values in the Additional information field of an Event that are not in JSON key/value format are normalized to JSON key/value format when the event is processed. For example, assume that the following plain text is in the Additional information field: &quot;Connection instance is successful&quot;. When the event is processed, all this plain text becomes one JSON string and might not be useful within an alert. In the resultant alert, this string is in the Additional information field in JSON key/value format, containing the data: {&quot;additional_content&quot;: &quot;Connection instance is successful&quot;}.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

You can respond to the alert in the following ways:

**Alert response options**

| Option                                    | Description                                                                                                                                                                                                 |
|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|
| Submit                                    | Save the modifications that were made to the form and return to the **Alerts** list.                                                                                                                         |  |
| Acknowledge the alert.                    | Click **Acknowledge**. If the alert is reopened, this button reappears so you can reacknowledge the alert.                                                                                                       |  |
| Create an incident.                       | Click **Create incident**. For more information, see **Create an incident or security incident from an alert**.                                                                                               |  |
| If Security Incident Response is activated, create a security incident response. | Click **Create Security Incident**.                                                                                                                                                                         |  |
| Designate that the alert is in maintenance.| Select the **Maintenance** check box. For more information, see **View all alerts by the maintenance status**.                                                                                            |  |
| Close the alert.                          | Click **Close**. For more information, see **Close an Event Management alert**.                                                                                                                             |  |

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Create an incident or security incident from an alert
When an alert must be escalated and assigned to someone who can resolve the underlying issue, you can open an incident.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

If Security Incident Response is activated, a security incident can be created.

You can manually create incidents and security incidents from the Alert form. To prevent duplicate tasks, the system checks the conditions of all task templates before creating an incident.

You can customize the created incident using the `EvtMgmtCustomIncidentPopulator.populateFieldsFromAlert` script include. The customization includes mapping fields from the alert to the incident or aborting the incident creation according to customized conditions. For more information, see Custom alert fields.

You can populate incident fields using custom alert fields values that were populated from additional information fields. Use the `EvtMgmtCustomIncidentPopulator` script include to copy the values to the incident after copying the data to the alert. For more information, see Custom alert fields.

**Note:** If Security Incident Response is activated, the base system includes an alert action rule called Create security incidents for critical alerts. This alert action rule creates security incidents when critical security events are reported.

1. Navigate to Event Management > All Alerts.
2. Click the alert Number.
3. To create an incident:
   - To create an incident, click Quick Incident.
   - To create a security incident, click Create Security Incident. You must install the Security (secops) plugin to enable this option.
4. Click Update.

The created incident appears in the Task field of the Alert form.

Resolve an incident related to an alert
When you resolve an incident that is associated with an alert, the alert can also close according to the `evt_mgmt.incident_closes_alert` property.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

When you close an incident by clicking Resolve Incident on an open incident, the corresponding alert is closed, resolved, or left alone as specified by the `evt_mgmt.incident_closes_alert` system property.

Launch web application from alert
You can launch a web application from an alert that matches the conditions set in an alert action rule.

Create an alert action rule that is configured to prepare a URL when the conditions that you stipulate are met. For example, based on the results when the alert parameters are resolved, this feature can open a Knowledge Base article or construct a URL.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

Right-click an alert in any screen to use this feature, for example, in the Alert Console, All Alerts, or the Alert list in the Event Management Dashboard, and choose Quick Response. Click the
required link to launch the web application that was configured in the alert action rule. For further information, see Migrate an alert action rule to an alert management rule.

1. Navigate to Event Management > Alerts Console.
2. Right-click the required alert and select Quick Response.
   A window appears with a list of available web applications that were configured in the alert action rule and the filter that matches the alert. To see the target URL, point to the cursor over the application name.

   **Note:** If one or more of the alert action rule parameters cannot be resolved, the related application name appears in black. When the alert parameters are resolved, then the application name becomes a link.

3. Click the name of an application to open the URL in another tab or window in your browser.

   A common use case is launch in context to the source management system.

   Other examples include to search in knowledge bases, not only within ServiceNow, but externally as well. Any URL-based action can utilize the alert parameters and the URLs can refer to wikis, messaging services, REST APIs, and so on.

Close an Event Management alert
Close an alert by an event or a user action. Closing an alert also closes any related incident that is not already resolved or closed.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

By closing an alert, any related incident that is not already resolved or closed is closed. This default behavior can be configured by changing the value of the evt_mgmt.alert_closes_incident property.

1. You can close an alert from the Alert form or the Event Management dashboard.
   - To close an event from the Alert form, navigate to Event Management > All Alerts. In the All Alerts list, click the required alert.
   - To close an event from the Event Management dashboard, navigate to Event Management > Dashboard. In the Alerts panel, click the required alert.

2. In the Alert Properties view, click Close.
   When you click Close, the alert is closed without confirmation.

   If the alert has any resolved or closed incidents, a work note is added to the incident indicating that the related alert was closed.

   If the alert has an open incident that is not related to any other open alerts, the incident is closed, resolved, or left alone as specified by the evt_mgmt.alert_closes_incident property.

Reopen an alert
Additional events can cause reopening of alerts, or you can reopen an alert by changing its state. When an alert reopens, any associated incidents can also be updated or reopened according to the incident state and the evt_mgmt.alert_reopens_incident property.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

When an alert is reopened, the related incident is processed as follows:

- If the incident is not Resolved or Closed, a work note is added to indicate that the related alert was reopened.
- If the incident is Resolved or Closed, the incident is reopened, a new incident is created, or nothing is done, depending on the evt_mgmt.alert_reopens_incident property value.
• If the incident is reopened, work notes are added to the incident.
• If a new incident is created, any matching alert management rule, alert action rule, and task template applies to the incident. If there is no matching alert action rule or template, fields from the existing incident are copied to a new incident.

1. Navigate to Event Management > All Alerts.
2. Search by alert number.
3. In the State field, click Reopen.
4. Click Update.

Apply a quick response in an alert
In an alert, use the Quick Response feature to apply remediation to the alert or to launch a web application.

Role required: evt_mgmt_operator, evt_mgmt_admin

Quick Response is a feature that enables you to perform a quick action on the selected alert. The action is to apply remediation or to launch an application. If you have upgraded from Istanbul or earlier, this feature is a combination of the Remediation and Launch an application features.

Right-click an alert, for example, in the Alert Console, All Alerts, or the Alert list in the Event Management Dashboard, and you can choose Quick Response to perform a quick action. If you select a remediation link, a workflow is executed according to the criteria specified in the alert management rule. Click a launch application link to launch the web application that was configured in the alert management rule. For further information, see Migrate an alert action rule to an alert management rule.

1. Navigate to Event Management > All Alerts.
2. In the Alerts list, right-click the required alert and select Quick Response.

Quick Response for Alert0110004

Click the appropriate link either to run remediation or to launch a web application.

Run Remediation

Remediation - Get Running Processes

Launch Application

Google
H!
ServiceNow docs

3. In the Quick Response page, click the required link to perform the required action. If the link that you require is not available, perform one of these actions:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| For Remediation, define a remediation task using an Alert rule based on the selected alert | 1. Navigate to Alert Rules.  
2. Select the required alert management rule to edit.  
3. Click Remediation.  
4. Click Enable Remediation.  
5. In the Execution field, select either Automatic or Manual.  
6. In the Orchestration workflow field, specify the remediation workflow.  
   - For Orchestration, in the workflow settings, select Remediation Task (em_remediation_task) in the Table field.  
   - After you finish configuring the orchestration workflow, publish it.  
7. Click Update.  
  
For further information, see Migrate an alert action rule to an alert management rule. |
| For Launch Application, create a web application link using an Alert rule based on the selected alert | 1. Navigate to Alert Rules.  
2. Select the required alert management rule to edit.  
3. Click Launcher.  
4. Click Enable.  
5. In the Display Name field, specify a name for the link.  
6. In the URL field, compose the URL using data from the alert in the format: ${source}.com:${port}/ ${cmdb_ci.name}  
7. Click Update.  
  
For further information, see Migrate an alert action rule to an alert management rule. |

To create a remediation task or web application link, see Migrate an alert action rule to an alert management rule.

**Alert intelligence**

Alert intelligence is implemented in the multi-tab interface of Agent Workspace. You can manage alerts in Workspace in the Event Management implementation of Agent Workspace.

**About alert intelligence in Workspace**

In Workspace, you can work on an alert record without navigating out of the interface. You can maintain context while opening and working in several tabs and respond to alerts without leaving the form. In the Workspace navigation pane, from the list of alerts sorted in order of smart priority, you can drill down to see key information relevant to the alert. For example, you can see the
priority of the alert, key indicators, and a time-line together with information from other alerts that are part of the alert group.

This implementation is intended for the Network Operations Center (NOC) operator. Its purpose is to enable the operator to resolve alerts in a more efficient and effective way. The operator can examine and diagnose alerts in an intuitive manner using insights (for example, past incidents, similar, or repeated alerts), access to relevant KB articles, graphic time-line of alert groups, and it enables the operator to perform actions (for example, create an incident) that can speed alert resolution.

Note: Do not activate the Alert Intelligence plugin manually. If this plugin must be activated, contact your ServiceNow account manager.

An alert is a notification generated by Event Management for selected events that are considered to be important and require attention. The generation of alerts is based on event rules. After alerts generate, the manner in which you can monitor and resolve alerts is based on alert management rules. If the alert generated an incident, you can click the incident and see the incident details.

For more information about Agent Workspace, see Agent Workspace.

Agent Assist is integrated into the Workspace interface. This feature enables you to perform contextual search, refer to relevant Knowledge Base articles, and save a link to the information found in the alert form. For more information about Agent Assist, see Using Agent Assist in Agent Workspace.

Browser support for workspaces

Do not use Internet Explorer 11 to access any workspaces. If you are on the New York release and you are using Internet Explorer 11 with any workspace, such as Service Owner Workspace, Vendor Manager Workspace, or any other type workspace, you must migrate to a modern browser before you upgrade to future releases. Internet Explorer 11 will not be supported for workspaces after the New York release. See KB0683275 for more information on Internet Explorer 11, and see Generally supported browsers for more information about browsers and what you can use across the platform.

Open Alert Intelligence in Workspace

The Event Management implementation of the Workspace interface can be reached in a number of ways.

Open Alert Intelligence in Workspace

Various ways to open Alert Intelligence in Workspace are:

- In the navigation pane, navigate to Event Management > Alert Intelligence.

- In a browser, add /workspace to the instance URL and press Enter.

- Open an alert in the Event Management Dashboard, the Alert Console, or the All Alert list. In the alert record that opens, click Open in Workspace.
From the right-click menu of an alert from the Event Management Dashboard, the Alert Console, or the All Alert list, select **Open in Workspace**.

**Navigate in Workspace**

When Workspace is open, to see how to navigate, using Agent Workspace as an example, see [Navigating Agent Workspace](#).

**Manage alert lists in Workspace**

Lists that are specific to Event Management alerts in Workspace are provided in the base system and are located in the Lists panel.
Alert lists in Workspace

Filtered lists

In the base system, Event Management provides lists that are filtered into various categories. Click the required filtered list to display the alerts that you need.

1. In the Workspace navigation bar, click .

   The list panel displays the categorized lists.

2. In the Lists panel, under Alerts, the filters provided in the base system are:

   Provided alert filters

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top Priority</td>
<td>List of open alerts that are in the highest Priority group category.</td>
</tr>
<tr>
<td>Top Severity</td>
<td>List of open alerts that have the highest Severity.</td>
</tr>
<tr>
<td>Open Alerts</td>
<td>List of open alerts.</td>
</tr>
<tr>
<td>Acknowledged Alerts</td>
<td>List of alerts whose Acknowledged field has been selected.</td>
</tr>
<tr>
<td>All Alerts</td>
<td>List of open and closed alerts.</td>
</tr>
</tbody>
</table>

Create a custom list
You can create a list that is filtered according to your requirements, and you can save the list using an easy-to-identify name.

1. In the Workspace navigation bar, click **My Lists**.
2. At the bottom of the Lists panel, click **New list**.
3. On the form, fill in the fields.

   **New List form**

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>List name</td>
<td>Name for the custom list.</td>
</tr>
<tr>
<td>Table</td>
<td>Database table from which the alert list fields are to be populated.</td>
</tr>
<tr>
<td>Conditions</td>
<td>Conditions that you define to select the required fields for the alert list.</td>
</tr>
<tr>
<td>Columns</td>
<td>Columns to be displayed in the alert list, chosen from the selected database table.</td>
</tr>
</tbody>
</table>

4. Click **Save**.
   In the Workspace navigation bar, the new custom filtered list appears in the **My Lists** pane.

**Alert list functions**

You can select one or more alerts on the alert list and click the relevant button on the top of the form:

<table>
<thead>
<tr>
<th>Button Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledge</td>
<td>Acknowledge the alerts to indicate that they require attention.</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Place the alerts in maintenance to indicate that the alert's CI is in the Maintenance state.</td>
</tr>
<tr>
<td>Close</td>
<td>Close the alerts.</td>
</tr>
</tbody>
</table>

**Update alerts in an alert list**

By default, alerts in the alert lists are automatically updated (auto-refresh) whenever an action is performed on an alert. You can configure the amount of time, in seconds, after which the alert list refreshes when an action is performed.

You can also disable auto-refresh and configure how often the alert list is to be updated. You may want to invoke this parameter when a large number of alerts are being processed continuously, so that the list is not constantly refreshing.

1. In the navigation pane, enter `sys_properties_list.do` in the search field.
2. To change the alert list refresh time, modify these properties:

   **Properties to configure alert list refresh**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>evt_mgmt.ai_refresh_time_to_wait_after_manual_action</code></td>
<td>Alert list refresh time after a manual action has been performed on the alert.</td>
</tr>
</tbody>
</table>
### Property and Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>evt_mgmt.ai_refresh_time_interval</code></td>
<td>Amount of time after which alert list refreshes, whether or not any action was performed on the alerts.</td>
</tr>
<tr>
<td><code>evt_mgmt.disable_lists_auto_refresh</code></td>
<td>Select to disable auto refresh of alerts in Workspace.</td>
</tr>
</tbody>
</table>

You can create a new list on the **Alert Lists Auto Refresh** page (enter `em_alert_lists_auto_refresh.list` in the navigation pane), and clear the **Enable Refresh** option so that the list does not refresh automatically.

### Open an alert from a list

You can open an alert from any of the list categories. For example, in the Lists panel, click **All Alerts** and then click the required alert.

### Actions you can perform in lists

From the list of alerts in Workspace, you can perform the following actions.

#### Search the list
Search just within the list you have open. The list search automatically searches all the columns in the list.

**Note:** If any records are hidden or removed due to security access restrictions, you see a message with the number of records that were hidden.

### Sort columns

Click the column headings to determine the sort order of a column. The arrow next to the column name indicates the current sort order. A downward pointing arrow indicates that the column is sorted in descending order. Only the primary sort order is indicated. The system remembers the most recent sort order that you choose for a specific column.

Column sorting follows the same default behavior as the platform. If a column is not currently sorted, it sorts ascending unless the column data type is a Date. Dates are sorted as descending by default.

### Change the number of rows per page

In the list footer, click the <Number> rows per page button and select the number of rows you want to display.

### Open a record

Click anywhere in the row to open a record. The list closes and the record opens in a new tab.

### Tab levels in an alert form

There are two levels of tabs in the alert form in the Workspace page.

#### Highest tab level

The alert number of the open alert displays in the highest tab level. The alert number of each subsequent alert that you open displays in its own tab. A maximum of 10 tabs can be opened.

#### Secondary tab level

In the alert form, click an object to open it in a secondary level tab. In the second level, you can have tabs that show, for example, details of the alert; related task (Incident) of the alert; related CI of the alert.
Alert state

If an alert is in **Closed** state, and is now changed to **Open**, after saving the alert and refreshing the alert record, the **Actions** button shows in the alert ribbon.

A user with the role of **evt_mgmt_admin** can change the state of an alert.

1. To change the state of an alert from Closed to Open, select the required alert in the alert list.
2. In the Details tab, in the State area, select Open and click **Save**.
   
   **Actions** now shows in the alert ribbon.

View a topology service map

You can view a topology map of all services affected by an alert. Click the **Actions** button for the alert and select **Dependency View**. For details on the topology map, see [View monitored services](#).

Create a major incident from an alert

When an alert meets the criteria set in the Alert Filter tab (on the legacy UI), you can create a major incident for the alert. Click the **Actions** button for the alert and select **Major Incident**. For details on managing major incidents, see [Major incident management](#).

View alert record details

To view the details of an alert record, click an alert in an alert list.

The following graphic depicts the details of an alert that is in Open state.
The areas on the alert form are described in the following table.

**Alert form areas**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A) Tab</td>
<td>Each record you click opens in a new tab within Workspace. A tab view of records makes it easier to navigate work in your queue.</td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>(B) Details tab</td>
<td>The Details tab includes form fields, where you can save and complete forms similar to the platform.</td>
</tr>
<tr>
<td></td>
<td>The header in the details tab includes these details:</td>
</tr>
<tr>
<td></td>
<td>• Description: The text from the Description field of the alert is displayed.</td>
</tr>
<tr>
<td></td>
<td>• Updated: Date and time when the alert was updated.</td>
</tr>
<tr>
<td></td>
<td>• Source: The text from the Source field of the alert is displayed.</td>
</tr>
<tr>
<td></td>
<td>• Overall Event Count: The total overall number of events related to this alert.</td>
</tr>
<tr>
<td></td>
<td>• Role in group (None/Primary/Secondary): The role that the selected alert has in the alert group.</td>
</tr>
<tr>
<td></td>
<td>• Group (None/CMDB/Manual/Automated): The group that the selected alert is a member of.</td>
</tr>
<tr>
<td>(C) Form</td>
<td>The following components make up a form:</td>
</tr>
<tr>
<td></td>
<td>• A header, which gives a quick glance of alert details and priority information.</td>
</tr>
<tr>
<td></td>
<td>• A ribbon, which shows additional alert information. The ribbon includes two widgets:</td>
</tr>
<tr>
<td></td>
<td>• Alert and Alert Group Timeline.</td>
</tr>
<tr>
<td></td>
<td>• The alert body, which gives a complete view of the alert information.</td>
</tr>
<tr>
<td></td>
<td>• Related lists, which show additional related information at the bottom of the form.</td>
</tr>
<tr>
<td>(D) Agent Assist attachment panel</td>
<td>Agent Assist displays relevant knowledge base (KBs) articles for the current alert to find information that might help in solving the alert. Use contextual search to find relevant knowledge base articles. Add the knowledge base articles as attachments to the alert, for example, supporting information to a customer issue.</td>
</tr>
<tr>
<td>(E) Details area</td>
<td>The details of the alert record are displayed in this area.</td>
</tr>
<tr>
<td></td>
<td>Included in the details is the Overall Event Count: The total overall number of events related to this alert. The Overall event count may be different when compared to the number of events, as the overall count includes historic events related to this alert but are no longer kept in the events table.</td>
</tr>
</tbody>
</table>

**Alert section of the information ribbon**

The Alert widget displays on the left-hand side of the information ribbon and is depicted in the following image.
Alert widget contents

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>Priority group category of the alert. For more information about priority groups, see Priority group.</td>
</tr>
<tr>
<td>CI or Node</td>
<td>The CI or node that is bound to the alert. Click the CI or node to open it in a sub-tab.</td>
</tr>
<tr>
<td>Severity</td>
<td>Display the severity status of the alert.</td>
</tr>
<tr>
<td>Actions</td>
<td>Displays actions that were defined in Alert Management. For more information about actions to resolve alerts, see Apply a quick response in an alert.</td>
</tr>
<tr>
<td>Feedback</td>
<td>Submit your feedback about the usefulness of the alert group. Feedback can help future analysis and, over time, accumulated feedback continues to improve the accuracy of alert aggregation.</td>
</tr>
<tr>
<td>Task</td>
<td>Open the underlying incident, change request, or problem. The task opens in a sub-tab.</td>
</tr>
</tbody>
</table>

Timeline section of the information ribbon

The Timeline widget, which displays on the right-hand side of the information ribbon, shows the Alert Group Timeline. For further information about the Alert Group Timeline, see Timeline information.

Collapse information ribbon

Click the to toggle the display of the alert and timeline ribbon.

Functions at the top of the alert form

Actions that can be performed in the alert form are listed in the following table.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insight</td>
<td>Retrieve and display Alert Insight details. For more information about Alert Insight, see View alert insight information in Workspace</td>
</tr>
<tr>
<td>Save</td>
<td>Update the selected alert.</td>
</tr>
<tr>
<td>Quick Incident</td>
<td>Create an incident that is linked to the selected alert.</td>
</tr>
<tr>
<td>Close</td>
<td>Close the selected alert.</td>
</tr>
<tr>
<td>Inform Customer Service</td>
<td>Create an internally reported case.</td>
</tr>
<tr>
<td></td>
<td>Toggle the display of the Agent Assist sidebar. For more information about Agent Assist, see Using Agent Assist in Agent Workspace.</td>
</tr>
</tbody>
</table>

Alert Management Executions

The entries in this section provide a reference to the actions that have been performed concerning alert management rules. For more information about alert executions, see Alert executions information.
Remediation Tasks

A remediation task is automatically created to capture every remediation that was applied to a CI or to an alert. For more information, see View remediation tasks.

Post comments in the Activity area

In the Activity area of alerts in Workspace, you can post comments or attach a solution.

1. In the Activity area of an alert, select Additional comments to make your entry visible to customers, or select Work notes to keep the entry internal.

2. In the selected comment field, write a comment or attach a solution and then click Post.

Priority group in Workspace

Alerts that require you to handle them at a higher priority than other alerts are brought to your attention by listing alerts with a higher priority at the top of the alert list. This leads to better triage and focus.

The priority group indicates which alerts should be attended to first. Priority is calculated for each open alert and then mapped into one of four priority categories.

The priority group of the selected alert is prominently displayed in the alert section of the ribbon. For example, the following graphic depicts that the selected alert has a High priority, indicating that it should be resolved ahead of alerts that have a lower priority.

For more information about priority groups, see Priority group.

Timeline information

The Alert Group Timeline tracks alert groups and displays them as points on a timeline.
Information displayed in the timeline

Each point in the timeline represents an alert at the time of the event generation.

Click an alert point to open the alert in a new tab.

If multiple alerts are generated at the same time, only the alert with the highest severity is displayed on the timeline.

An alert point on the timeline that is surrounded by a blue frame indicates that it represents the current alert.

Hover over an alert point to show a tooltip that includes these details:

- CI or Node
- Alert number
- Event Generation Time
- Alert priority group category
Filter alerts on the timeline

Click **Show legend** to display the timeline legend and filter. The color of the alert point represents the severity of the alert.

To filter out the display of alerts on the timeline, in the legend, click the alert point type that is not required to be displayed.

To close the display of the timeline legend, click **Hide Legend**.

Zoom into a time range

To focus on the time range of an alert point, on the timeline you can zoom in the required area. Zoom by dragging the mouse over the required section of the timeline.

Release the mouse to show the time range on the timeline. The highlighted area is brought into focus.
To return the timeline to its previous state, click ![Reset zoom](image).

**Alert insight information**
Alert insight aids faster alert triage, enabling a quicker way to find a solution and expose the probable root cause of the selected alert.

**Use alert insight to find a solution and help find the root cause of the alert**

Results from data analysis provide further information about the alert, beyond that of the alert form. This aids faster alert triage, enabling a quicker way to find a solution and expose the probable root cause of the alert.

Information shown is from repeated alerts, similar alerts, incidents, problems, and change requests on the same CI, similar CIs, and other CIs that are related by CMDB relations and business service (BS) relations, or grouping. For more information about the alert insight tabs in the alert form, see [View alert insight information](#).

**View alert insight information in Workspace**
View insight information on the alert and the CI based on history, analysis of CMDB, and ITSM data.

Role required: evt_mgmt_user for alert insight information and evt_mgmt_admin for all actions, for example, resolve the alert.

Alert insight provides further information about the alert, for example, to show how similar alerts were resolved and provide further insight into the root cause of the alert. View relevant data in Workspace in the context of the alert and the CI as well as in the context of related alerts and CIs.

The various sections in this feature contain insight data analysis on the alert and the related CIs based on history, analysis of CMDB, and ITSM data. You can, for example, see repeated or similar alerts and this information can help you to investigate problems further. The information in the **Repeated alerts** tab is relevant, while the information in all the other tabs provide suggestions that might be relevant to the selected alert.

**Note:** Secondary alerts that have any of the following settings are not displayed:
- Severity is Info
- State is Closed
- Maintenance is selected.
Domain separation is supported. For more information, see Domain separation and Event Management.

1. In Workspace, navigate to List > All Alerts.
2. Click the required alert.
3. Optional: You can click **Actions**.

Select an action to resolve the alert, such as Launch Application, or Remediate.
- If you select a remediation link, a workflow is executed according to the criteria specified in the alert management rule.
- If you click launch application link, it launches the web application that was configured in the alert management rule.
- If you clicked subflow, the selected subflow is executed according to the criteria specified in the alert management rule.

4. The Alert Group Timeline depicts the time of occurrence of the alert in the group it belongs to. Alert groups are displayed across a timeline according to the update time of the alerts. For more information, see Timeline information.

5. To view in-depth alert analysis, in the Details panel, click Insight.
The alert insight details open on the second-level row of tabs. Subsequent actions are displayed on this level as well.

6. Review the alert insight information on the selected alert record.
In each alert insight block, drill down to see if the solution or root cause can be determined.

- View current CI and similar CIs to find the most suitable solution to remediate and resolve this alert.
- Probable root cause is based on machine learning analysis. Review results to find the top Incidents, Changes, and Problems for the current CI. For related CIs, find the most likely probable root cause for the selected alert.

### Alert insight in Workspace

<table>
<thead>
<tr>
<th>Alert insight block</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Top Alerts</strong></td>
<td></td>
</tr>
<tr>
<td>Repeated alerts</td>
<td>List of alerts that have the same message key as the insight alert CI and the alerts which are in closed state. Use this insight information to find a solution for the alert. Timeframe: Alerts that are up to 30 days old are retrieved.</td>
</tr>
<tr>
<td>Similar alerts</td>
<td>List of similar alerts that have the same CI type as the insight alert CI, as well as the same source, resource, metric name, and so on. Use this insight information to find a solution for the alert and to see similar alerts that were resolved. Timeframe: Alerts that are up to 30 days old are retrieved.</td>
</tr>
<tr>
<td><strong>Top Incidents</strong></td>
<td></td>
</tr>
<tr>
<td>Current CI</td>
<td>List of incidents that have the same CI as the insight alert CI. Timeframe: Incidents that were created up to 7 days before the alert was created are retrieved.</td>
</tr>
<tr>
<td>Related CIs</td>
<td>List of incidents on CIs that have some level of relationship with the insight alert CI. Use this insight information to find the root cause of the alert. Timeframe: Incidents that are up to 7 days old from the creation time of the alert are retrieved.</td>
</tr>
<tr>
<td><strong>Top Changes</strong></td>
<td></td>
</tr>
<tr>
<td>Current CI</td>
<td>List of change requests on CIs that have a level of relationship with the insight alert CI. Use the insight information to find the root cause of the alert.</td>
</tr>
<tr>
<td>Related CIs</td>
<td>List of change requests that have the same CI as the insight alert CI. Change requests are only retrieved for these states: implement, closed, and review. Use this insight information to find the root cause of the alert. Timeframe: Incidents that were created up to 7 days before the alert was created are retrieved.</td>
</tr>
<tr>
<td><strong>Top Problems</strong></td>
<td></td>
</tr>
<tr>
<td>Current CI</td>
<td>List of problems that have the same CI as the insight alert CI. Use this insight information to find the root cause of the alert. Timeframe: Incidents that were created up to 7 days before the alert was created are retrieved.</td>
</tr>
<tr>
<td>Alert insight block</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Related CIs</td>
<td>List of problems on CIs that have a level of relationship with the insight alert CI. Use this insight information to find the root cause of the alert. Timeframe: Problems that are up to 7 days old from the creation time of the alert are retrieved.</td>
</tr>
</tbody>
</table>

Use properties to configure alert insight. For more information, see [Alert insight properties](#).

**Alert correlation rules**

Alert correlation rules enable you to manually classify alerts into primary and secondary, and establish a relationship between them. Use alert correlation rules to group alerts that are related.

For example, if a server in your organization goes offline, you do not need to see additional alerts that show that the virtual machines or applications running on the server are also down. Instead, these additional alerts, or secondary alerts, are grouped under the root alert called the primary alert. You can see these grouped alerts in the alert list in Alert Intelligence.

**Note:** Event Management alert aggregation and RCA provides alert groups for an automated correlation of events and alerts. These alert management rules give you full manual control of alert correlation. For more information on the alert aggregation and RCA feature, see [Automated alert groups](#).

**Primary and secondary alerts**

- A primary alert identifies the root cause of an event.
- A secondary alert is another alert that the same event generates, but is considered less important than the primary alert.

The purpose of specifying secondary alerts is to identify which alerts to suppress, so you can reduce the amount of alert noise and focus on the primary alert. You are able to see both types of alerts, but the secondary alerts are grouped under the primary alert. You can see these groups in the alert list in Alert Intelligence.

**How primary and secondary alert management rules work**

Primary and secondary alert management rules are simply filters that specify which kinds of alerts are classified as primary and which kinds are classified as secondary. The filter criteria runs against the Alert (em_alert) table.

An alert can belong to more than one group, but only one level of secondary alerts is allowed under a primary alert. For example, if secondary alert A is grouped under secondary alert B, both alerts become secondary alerts under the same primary alert. See [alert hierarchy](#) for more information.

**Alert relationships**

Alert correlation rules enable you to specify what type of relationship the primary and secondary alerts must have for the rule to match. These relationships depend on the relationships between the CIs in the CMDB:
- **No Relationship**: Ignore the relationship when looking for a match.
- **Same CI or Node**: Relate both alerts with the same CI. If the CI field is blank, then the alerts must have the same Node value.
- **Primary is Parent**: The relationship is in the direction of parent (primary) to child, as described in the CI Relationship Types table (cmdb_rel_ci).
- **Primary is Child**: The relationship is in the opposite direction, child (primary) to parent, as described in the CI Relationships table (cmdb_rel_ci).

### When correlation rules run

Alert correlation rules run when an alert is created or reopened. The alert is checked to determine if it matches being primary or secondary.

When the primary alert severity is changed to **Closed**, the secondary alerts are also set to **Closed** unless they are part of other groups where the primary alert is not yet set to **Closed**.

### Alert hierarchy

Only one level of secondary alerts is permitted. In situations where a secondary alert has its own secondary alert, the Event Management application flattens the hierarchy to preserve only two levels.

For example, assume alert A is the primary alert and alert B is the secondary alert. If alert C becomes a secondary alert for alert B, the application flattens the hierarchy so that A remains the primary and B and C become sibling secondary alerts, one level below A.

As another example, assume that there are three correlation rules that produce the following results:

1. Rule 1 (with an Order value of 1): B becomes a primary alert for A.
2. Rule 2 (with an Order value of 2): A becomes a primary alert for C and D.
3. Rule 3 (with an Order value of 3): E becomes a primary alert for A.

When alerts B, C, D, and E are triggered, they all appear in the alert list separately because there are no correlations between them.

When alert A is triggered:

1. Rule 1 makes A a secondary alert under alert B.
2. Rule 2 makes both C and D secondary alerts under alert A.
3. Rule 3 makes A a secondary alert under E, giving alert A two primary alerts above it in the hierarchy.

Therefore, if all alerts are triggered, only B and E appear in the alert list, indicated as primary alerts.

**Note**: A secondary alert can be correlated with more than one primary alert, and vice versa.

### View correlated alerts

You can view primary alerts on the alert console and see correlated alerts on a primary alert record.
Role required: evt_mgmt_admin or evt_mgmt_operator

1. Navigate to Event Management > Alert Console. The Group column contains an icon indicating that an alert is a primary alert that has one or more secondary alerts correlated with it.

2. Ensure that Correlated Alerts is selected.

3. Click the primary alert number to open the record for the primary alert. The Correlated Alerts section shows a list of all the secondary alerts that are correlated with this primary alert.

Correlated alert example

Create an alert correlation rule
Create an alert correlation rule to specify a primary alert and a related alert that is of secondary importance.

Role required: evt_mgmt_admin

Note: If you delete an alert correlation rule, the existing correlation groupings on the alert console are not removed.

2. Click New.
3. On the form, fill in the fields.
## Alert Correlation Rule form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name to identify the correlation rule.</td>
</tr>
<tr>
<td>Order</td>
<td>The priority for rule evaluation. Rules with lower-order values are given priority. An alert is checked against every alert action rule until a match is found.</td>
</tr>
<tr>
<td>Active</td>
<td>Select to activate the rule.</td>
</tr>
<tr>
<td>Advanced</td>
<td>Select to display the script field. This option enables you to script the event correlations.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the rule.</td>
</tr>
<tr>
<td>Primary Alert</td>
<td>The filter condition to identify the alert that is the primary alert, or most important alert, in a set of related alerts.</td>
</tr>
<tr>
<td></td>
<td>This field does not appear when Advanced is selected.</td>
</tr>
<tr>
<td>Secondary Alert</td>
<td>The filter condition to identify the alert that is related to the primary alert, however it is of lesser importance.</td>
</tr>
<tr>
<td></td>
<td>This field does not appear when Advanced is selected.</td>
</tr>
<tr>
<td>Filter</td>
<td>The filter condition to identify the alert on which the script is run.</td>
</tr>
<tr>
<td></td>
<td>Filter is available only when Advanced is selected.</td>
</tr>
<tr>
<td>Relationship Type</td>
<td>Specify the type of relationship between the primary and secondary alert:</td>
</tr>
<tr>
<td></td>
<td>- No Relationship: Ignore the relationship when looking for a match.</td>
</tr>
<tr>
<td></td>
<td>- Same CI or Node: Relate both alerts with the same CI. If the CI field is blank, then the alerts must have the same Node value.</td>
</tr>
<tr>
<td></td>
<td>- Primary is Parent: The relationship is in the direction of parent (primary) to child, as described in the CI Relationship Types table (cmdb_rel_ci)).</td>
</tr>
<tr>
<td></td>
<td>- Primary is Child: The relationship is in the opposite direction, child (primary) to parent, as described in the CI Relationships table (cmdb_rel_ci)).</td>
</tr>
<tr>
<td></td>
<td>This field does not appear when Advanced is selected.</td>
</tr>
<tr>
<td>Time Difference in Minutes</td>
<td>The minutes between which the primary and secondary event must occur to match this rule. The default value is 60 minutes.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The value for this entry cannot exceed 1440 minutes (one day).</td>
</tr>
<tr>
<td></td>
<td>This field does not appear when Advanced is selected.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Script</strong></td>
<td>Custom script that you can modify to return a JSON string that specifies the primary and secondary alerts. Select Advanced to display the script field.</td>
</tr>
<tr>
<td></td>
<td>/* The function needs to return a JSON-</td>
</tr>
<tr>
<td></td>
<td>{correlationType:[correlatedAlerts]} for example: if your filter matches the alert, set the alert as the primary alert and set alerts 1, 2 and 3 each as secondary alerts.</td>
</tr>
<tr>
<td></td>
<td>You can use both multiple primary alerts and multiple secondary alerts.</td>
</tr>
<tr>
<td></td>
<td>The correlationType can be PRIMARY or SECONDARY, and the alerts ID must be in an array.</td>
</tr>
<tr>
<td></td>
<td>CurrentAlert is the GlideRecord of the currentAlert on which that rule runs.</td>
</tr>
<tr>
<td></td>
<td>The system supports only one primary per alert, so:</td>
</tr>
<tr>
<td></td>
<td>Do not correlate more than one alert under the PRIMARY array.</td>
</tr>
<tr>
<td></td>
<td>Do not correlate alerts that already have a primary under the SECONDARY array.</td>
</tr>
<tr>
<td></td>
<td>The system supports open alerts only, so do not correlate alerts that have been closed under either one of the arrays.</td>
</tr>
<tr>
<td></td>
<td>*/</td>
</tr>
<tr>
<td></td>
<td>(function findCorrelatedAlerts(currentAlert){</td>
</tr>
<tr>
<td></td>
<td>var result = {}; //Insert your code here</td>
</tr>
<tr>
<td></td>
<td>result = {'SECONDARY':</td>
</tr>
<tr>
<td></td>
<td>['alertID1','alertID2','alertID3']};</td>
</tr>
<tr>
<td></td>
<td>return JSON.stringify(result);</td>
</tr>
<tr>
<td></td>
<td>})(currentAlert);</td>
</tr>
<tr>
<td><strong>Relationship</strong></td>
<td>Description of the CI relationship between primary and secondary, for example, Allocated from::Allocated to or Allocated to::Allocated from. This field displays only if either Primary is Parent or Primary is Child is selected for the Relationship Type.</td>
</tr>
<tr>
<td><strong>Relationship Type</strong></td>
<td>Primary is Child</td>
</tr>
<tr>
<td><strong>Relationship</strong></td>
<td>Allocated from::Allocated to</td>
</tr>
</tbody>
</table>

4. Click Submit.
An alert group is created when a new alert was generated or the status of an alert changed from close or flapping to open or reopen and the filter is matched.

**Alert aggregation and RCA**

The alert aggregation and root cause analysis (RCA) capability enhances Event Management with alert data analysis and alert aggregation. It also provides root cause analysis (RCA) for automated alert groups, and CMDB alert groups.

Use alert aggregation and RCA to do the following:

- Aggregate alerts to create **automated alert groups**.
  
  Correlate alerts according to timestamps and CI identification to create automated alert groups. Alert correlation helps organize incoming real-time alerts and reduce alert noise.
- Apply RCA to automated alert groups and to CMDB alert groups.
- Correlate alerts based on CIs' relationships in the CMDB to create **CMDB alert groups**.
- Generate a pattern for a **manual alert group** and then create an automated alert group according to that pattern.

**Alert aggregation**

Alerts are grouped based on the CI that is associated with the alerts. Event Management groups alerts that are similar, but not necessarily identical, and also based on how close in time the alerts were created.

Alert aggregation has these components:

**Alert Aggregation Learner**

An offline job that runs once a day to process past alerts. The Alert Aggregation Learner identifies patterns of related alerts using a combination of pattern-based and probabilistic techniques. If the CMDB property used for grouping alerts in alert aggregation (**sa_analytics.agg.learner_group_by_property**) is set, then before processing starts, the Alert Aggregation Learner groups alerts by the specified CMDB property. For example, setting the property's value to **location** indicates that the property takes the CMDB location into account when grouping alerts.

**Note:**

- When configuring this property, ensure that you use the column name (in this example, **location**), and not the column label (**Location**).
- Configuring this property causes patterns to be created only for alerts within individual groups, and not across groups. You must ensure that the CMDB property used for grouping alerts in alert aggregation (**sa_analytics.agg.learner_group_by_property**) does not contain overly unique values (such as **name**, **sys_id**, and so forth. If it does, the number of alerts created in a group is limited, which prevents the job from creating a pattern.

**Real-Time Query**

A scheduled job that runs every minute and updates alert aggregation groups. It tries to match real-time alerts with alert patterns stored in the alert knowledge base.

**Aggregation reasoning work notes**
Work notes in the alert display the reason for the creation of the alert group. For example, if the alert was created due to a pattern, a link to the pattern is given.

If the `evt_mgmt.alert_groups_reasoning.enable_worknotes` property is set to `none`, work notes are not written.

RCA for automated alert groups

If you set the Enable root cause analysis for automated (pattern-based) groups property (`sa_analytics.agg.learner_rca_detection`) to `true`, Event Management applies RCA algorithms to automated alert groups. RCA identifies the root cause alert within the automated alert group. RCA helps to direct resources to the root cause CI of a problem. Root cause alerts are displayed in the timeline of alerts in Alert Intelligence.

RCA for CMDB alert groups

If the Enable CMDB Correlation for Alert Aggregation (`sa_analytics.agg.query_cmdb_correlation_enabled`) property is set to `true` and the Enable root cause analysis for CMDB groups (`sa_analytics.agg.query_cmdb_rca_enabled`) property is set to `true`, Event Management applies RCA algorithms to CMDB alert groups. RCA identifies the root cause alert within the CMDB alert group. RCA helps to direct resources to the root cause CI of a problem. Root cause alerts are displayed in the timeline of alerts in Alert Intelligence.

Specify and manage pattern identifier attributes for alert aggregation

Alert aggregation learns the alerts and then forms patterns based on a set of alert and CI attributes. You can specify the set of CI and alert attributes that will be used as the pattern identifier attributes for learning patterns, that will result in alert groups that are meaningful in your environment.

The default attribute used for forming patterns is `metric_name`. Navigate to Event Management > Alert Aggregation and RCA > Manage Pattern Identifier to view which pattern identifier attributes are currently in effect, to choose a different set of attributes to deploy, or to define a new set of pattern identifier attributes.

To ensure that the specified pattern identifier attributes used for forming patterns is effective, a sufficient number of alerts must have the respective attributes populated. Therefore, if you specify a new set of pattern identifier attributes, do the following to ensure meaningful analysis:

- Create an event rule that populates the respective attributes.
- If a large number of existing alerts do not have values for the new set of pattern identifier attributes, ensure to run the Service Analytics Attribute Populator for Historical Alerts job which uses the appropriate event rule to populate attributes in historical alerts. Properties originating from the CMDB CI using dot walking are not populated.
- Choose effective identifiers:
  - Ensure that the set of pattern identifier attributes is not too unique (for example, the date field is unique for every alert), because it will be impossible to identify any pattern.
  - Ensure that the set of pattern identifier attributes is not too common, because it will not be possible to create distinct groups, as everything would be included in the same group.

When an alert pattern is discovered based on a set of identifier attributes, the alerts are considered to be related to each other and therefore are grouped together into a Learned Pattern. For example, if you configure identifier attributes to create a pattern based on alerts with the same Priority Group and Resource, if a group of alerts match those attributes, they are
grouped into a pattern which displays on the Learned Patterns report (Event Management > Alert Aggregation and RCA > Learned Patterns).

Only one set of pattern identifier attributes can be active at a time. A new set of pattern identifier attributes is not automatically implemented until you deploy it. When you deploy a new set of attributes, the current set of attributes that is in effect becomes inactive. Subsequent queries use the active pattern identifier attributes to perform alert aggregation.

1. Navigate to Event Management > Alert Aggregation and RCA and then click Manage Pattern Identifier.
2. On the SA Alert Aggregation Pattern Attributes page click New.
3. Click the Feature Identifier Attributes icon and move attributes from the Available list to the Selected list.
   The Configuration Item attribute is automatically included as an identifier with all selections.
4. Click Submit.
5. Optionally, activate the newly specified pattern identifier attributes by selecting the pattern identifier attributes that you want to activate and clicking Deploy.

After specifying a new set of pattern identifier attributes, ensure that a matching event rule exists that populates the respective alert attributes. Also, run the Service Analytics Attribute Populator for Historical Alerts job to populate the respective attributes of historical alerts, if values are missing in existing alerts.

Automated alert groups

Event Management alert aggregation and RCA aggregates alerts into automated alert groups that represent the underlying event data. Automated alert groups are displayed in the Alert Console and in the Event Management dashboard.

To enable creating automated alert groups, set the Enable alert aggregation (sa_analytics.aggregation_enabled) property to true. For troubleshooting, ensure that the Service Analytics group alerts using RCA/Alert Aggregation scheduled job is running.

If Domain Support - Domain Extensions Installer is activated, then alert aggregation is applied at the domain level that is specified in the sa_analytics.agg.learner_domain_level property. By default, this property is set to 2, which is the second domain level in the domain hierarchy.

To create automated alert groups, aggregation algorithms rely partly on historical data in the alert knowledge base from similar past alerts and processes. As alerts continue to be generated and processed, data is collected and incorporated into the alert knowledge base for future processes and analysis.

Automated alert groups provide these capabilities:

Predictive alerts

If an automated alert group pattern contains at least four alerts, then alert aggregation uses the information of these existing alerts to generate predictive alerts. Each predicted alert is associated with a probability percentage number which represents the confidence in this alert actually being generated in the future. Predicted alerts with probability above the value of the Alert Prediction Minimum Confidence Score (%) Threshold property, are then displayed in the Group Timeline view in the Alert Console.

Root Cause Analysis

Alert aggregation applies Root Cause Analysis (RCA) to automated alert groups, to identify a root cause alert within the group. Identified root cause alerts are then displayed with a star, in the Group Timeline view in the Alert Console. If a root cause alert is identified for an automated alert group, then that alert is designated as the primary alert of the group.
View automated alert groups

View automated alert groups from the Event Management dashboard, the Alert Console, or a business service map. Automated alert groups are displayed in a list view, and are included in the timeline view in the Alert Console. When displaying an automated alert group that is associated with services, the impacted services are displayed and you can view the root cause CI if applicable.

Role required: evt_mgmt_user

In the Alert Console, from each automated alert group in the list view or in the timeline view, you can drill down to view all the alerts that are included in that group. The overall severity of an automated alert group corresponds to the highest alert severity within the alert group. The color that is associated with a automated alert group represents its severity.

The Event Management dashboard does not let you drill down to view further details of an automated alert group.

1. To display all automated alert groups:
   a) Navigate to Event Management > Alert Console.

In the list view, automated alert groups are noted by an icon in the Group column, and the value is Automated.

b) Click the icon for a parent alert to open the hierarchical list of alerts for an automated alert group.

For details about the alert form, see View alert information

2. To display automated alert groups that are associated with the services displayed on the Event Management dashboard:
   a) Navigate to Event Management > Dashboard.
   b) Enable Correlated Alerts.
   c) In the list view, automated alert groups are noted by an icon in the Group column.

- Click on a parent alert to open its alert form and provide feedback about the accuracy and helpfulness of the automated alert group.
- Add or remove alerts from the group.
- View root cause alert for an automated alert group.

View predicted alerts

Alert correlation analyzes existing alerts in an automated alert group, to predict future alerts for that group. Predicted alerts can indicate future problems with the associated services. By viewing those predicted alerts in the Alert Console, you can take measures to prevent any anticipated problems.

Enable creating predicted alerts by setting the (sa_analytics.pred.cl_enabled) property to true by performing the following steps:
1. Navigate to Event Management > Alert Aggregation and RCA > Properties.
2. Select the Enable alert prediction check box.
If necessary, change the default threshold for creating predicted alerts, which is controlled by the (sa_analytics.pred.cl_min_conditional_probability) property:

1. Navigate to Event Management > Alert Aggregation and RCA > Properties.
2. Change the value in the Confidence score threshold (%) after which alert prediction is generated field.

Predicted alerts are generated only if an automated alert group pattern contains at least four alerts and each of these alerts occurs on at least two CIs. Only the three predicted alerts with the highest probability numbers are displayed, all of which must be higher than the value of the property that controls the threshold for creating predicted alerts. The default threshold for creating predictive alerts is 10. Predicted alerts are distinguished from actual alerts by the color gray.

1. Navigate to Event Management > Alert Console.
2. Enable the Group Timeline view.

Only alerts for which probability meets or exceeds the threshold value are displayed.

Predicted alerts are displayed as gray dots on the group timeline, along with actual alerts. You can point to a predicted alert to display details such as the probability that was calculated for this alert, and the CI it is associated with.

View root cause alert for an automated alert group

If root cause analysis (RCA) is applied to automated alert groups, then you can view the identified root cause alert within the group. This helps identify the root cause CI when troubleshooting a problem.

Ensure that the (sa_analytics.agg.learner_rca_detection) property is set to true by performing the following steps:

1. Navigate to Event Management > Alert Aggregation and RCA > Properties.
2. Select the Enable root cause analysis for Automated (pattern-based) groups check box.

To enable RCA for automated alert groups.

1. Navigate to Event Management > Alert Console.
2. Enable the Group Timeline view.
3. Identify the automated alert group for which you want to view root cause alert.

In the list view, automated alert groups are noted by an icon in the Group column, and the value is Automated.

4. Click to open the hierarchical list of alerts in the group.

Root cause alerts are noted as a star on the timeline.

CMDB alert groups

Alerts are grouped by Event Management alert aggregation and root cause analysis (RCA) using different methods of correlation. For CIs without historical data, alerts are correlated based on those CIs relationships in the CMDB. CMDB alert groups are displayed in the alert list in Alert Intelligence and in the Event Management dashboard.

To correlate alerts into groups, alert aggregation and RCA learns from historical alert data and then forms alert patterns. Alert aggregation and RCA then attempts to match new alerts with these patterns to correlate alerts and create alert groups. However, in some situations, such as,
with a new implementation, or with a new set of CIs, there is no historical data to learn from.
In these situations, alert aggregation and RCA can automatically correlate alerts based on CI relationships. This correlation is based on hosting rules, containment rules, and suggested relationships. For example, the alerts for the CIs in the following relationships can be correlated into a CMDB alert group:

- A server hosting a computer
- Processes that are running on a specific server

**Note:** The hosting and containment relationships that are used for CMDB-based grouping are used only if the number of connections between the CIs is small. If two CIs are related through many connections, the connection is considered to be too weak for CMDB-based grouping.

You can view all alert groups by navigating to Event Management > Alert Intelligence. The icon in the Group column denotes the alert group type. Alerts that do not have an entry in the Group column are not correlated with any group.

If your ServiceNow instance uses domain separation, domain names are considered when forming groups.

**RCA for CMDB alert groups**

Alert aggregation and RCA apply RCA to identify a root cause alert within the CMDB alert group if the following properties are set to true:

- Enable CMDB Correlation for Alert Aggregation ([sa_analytics.agg.query_cmdb_correlation_enabled](#))
- Enable root cause analysis for CMDB groups ([sa_analytics.agg.query_cmdb_rca_enabled](#))

If the `true` property is set to true and CMDB alert groups are forming, then identified root cause alerts are then displayed with a star in the Alert Group Timeline view in Alert Intelligence.

**RCA indicator on alerts timeline**

If a root cause alert is identified for a CMDB alert group, that alert is designated as the primary alert of the group.

**Configure automatic creation of CMDB alert groups**

Use the properties listed in this table to control which alerts are automatically included in CMDB alert groups. For more information about Event Management properties, see Components installed with Event Management.
Properties to control CMDB alert groups

<table>
<thead>
<tr>
<th>Property</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable CMDB Correlation for Alert Aggregation</td>
<td>sa_analytics.agg.query_cmdb_correlation_enabled&lt;br&gt;Enable to allow alert aggregation and RCA to automatically use CI relationships to correlate alerts and form CMDB alert groups.</td>
</tr>
<tr>
<td>Enable Suggested Relations for CMDB Correlation</td>
<td>evt_mgmt.related_cis_use_suggested_relations_rules&lt;br&gt;Enable to use any suggested relationship that is defined in the system when forming CMDB alert groups.</td>
</tr>
<tr>
<td>CMDB Groups: Relationship level</td>
<td>sa_analytics.agg.query_cmdb_graph_walk_nodes&lt;br&gt;Set the number of levels to use for dot-walking. Dot-walking provides access to fields on related tables from a form, list, or script. For more information, see Dot-walking examples.&lt;br&gt;The setting for this property impacts the application of CMDB hosting rules, containment rules, and endpoints to CMDB group formation during alert aggregation.</td>
</tr>
<tr>
<td>evt_mgmt.related_cis_use_containment_rules</td>
<td>to disable CMDB alert groups from forming when using hosting and containment relationships. To add this property to your instance, navigate to System Properties &gt; All Properties and click New. Specify these details:&lt;br&gt;- Name: sa_analytics.agg.query_cmdb_containment_enabled&lt;br&gt;- Type: true</td>
</tr>
<tr>
<td>sa_analytics.agg.ignore_cmdb_applicative_flow</td>
<td>to prevent CMDB groups from forming due to applicative flow relations. To add this property to your instance, navigate to System Properties &gt; All Properties and click New. Specify these details:&lt;br&gt;- Name: sa_analytics.agg.ignore_cmdb_applicative_flow&lt;br&gt;- Type: true</td>
</tr>
<tr>
<td>sa_analytics.agg.query_cmdb_rca_enabled</td>
<td>to cause analysis for CMDB groups. This property is enabled by default.&lt;br&gt;To access this property, navigate to Event Management &gt; Alert Aggregation and RCA &gt; Properties. Then select the Enable root cause analysis for CMDB groups check box.</td>
</tr>
</tbody>
</table>

View root cause alert for a CMDB alert group
View the identified root cause alert within the group to identify the root cause CI when troubleshooting a problem.

Ensure that the (sa_analytics.agg.query_cmdb_rca_enabled) property that supports root cause analysis for CMDB groups is enabled:

1. Event Management > Alert Aggregation and RCA > Properties.<br>2. Ensure that the Enable root cause analysis for CMDB groups check box is selected.

The Enable CMDB Correlation for Alert Aggregation property must be set to true to enable the formation of CMDB alert groups.
Role required: evt_mgmt_admin

1. Navigate to Event Management > Alert Intelligence.

2. In the Lists pane, click All Alerts.

<table>
<thead>
<tr>
<th>Number</th>
<th>Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert0010002</td>
<td>Automated</td>
<td>Spectrum: RadWare CHASSIS DOWN IP:10.0.0.0</td>
</tr>
<tr>
<td>Alert0010003</td>
<td>CMDB</td>
<td>Demo Hardware hosting</td>
</tr>
<tr>
<td>Alert0010004</td>
<td>None</td>
<td>monitor value: 5</td>
</tr>
</tbody>
</table>

3. Click the alert that has the CMDB group for which you want to view root cause alert. For the purposes of example, Alert0010003 is opened.

Root cause alerts are noted as a star on the timeline. The root cause of the alert can be viewed on the timeline.

**Root cause CIs for business services**

For alerts associated with discovered business services and application services, Event Management alert aggregation and RCA can apply root cause analysis (RCA). RCA identifies the CI that is the underlying root cause in a business service.

If Domain Support - Domain Extensions Installer is activated, then RCA is domain aware. Alerts are analyzed within the context of the domain that business services or application services belong to. RCA for a business service or application service runs on the MID Server that is in the
same domain as the business service or application service. If there is no MID Server for a specific domain, then the MID Server from the global domain is used.

**Root cause analysis**

The ongoing operations of an application can generate many events and alerts which can become overwhelming when problems arise. If the system is experiencing a problem, the application service of assessing the impact of the alerts and identifying the underlying cause might require extensive resources and be lengthy.

To identify underlying problems, RCA algorithms prioritize alerts, group them in the context of impacted services, and identify *root cause CIs*. The root cause CI is a CI in a business service from which the root alert for an incident originated and which subsequently triggered additional alerts.

RCA has these components:

**RCA Learner**

An offline job that runs once a day to process past alerts. It collects information about frequent alert patterns within a service context, and stores this information in the alert knowledge base. Based on past alerts and on the impact model, the RCA Learner creates a probability model that can be used to answer cause and effect queries.

**Real-Time Query**

A scheduled job that runs every minute to group alerts and to update root cause CIs. It queries past lists of root cause recommendations to get the probability score for real-time alerts associated with other open alerts within the service.

For discovered business services and for application services, the Learner collects and analyzes data from past alerts. For new alerts, the Learner applies existing knowledge from similar past alerts, and continues to capture and analyze data from new alerts. As more alerts are encountered and resolved, the alert knowledge base grows and the precision of diagnosing the root cause CI improves.

Business services are discovered by Service Mapping and are represented internally in the system by a service model. The service model of the business service is used for identifying CIs related to the root cause CI.

When the root cause CI in a service is known, operators can create a single incident ticket and engage only the needed IT operator to expedite remediation. The IT operator can direct troubleshooting efforts to remove the root cause problem, and stop the recurrence of undesirable events.

**RCA-related properties**

By default, RCA is not applied for business services and for application services. You can enable RCA and modify other RCA-related behavior by changing the settings of the `sa_analytics.aggregation.include_service` and `sa_analytics.rca_enabled` properties.

**RCA configurations**

RCA uses an RCA configuration that filters and scopes the alerts to be analyzed. The base system includes pre-defined RCA configurations which might not be optimal in every environment. See RCA configurations for more information.
Confidence score

To help you decide how to invest troubleshooting efforts, RCA algorithms calculate a confidence score for the identified root cause CI. The confidence score is based on the Learner data and expresses the confidence in the identification. For example, a confidence score of 75% means that there is a certainty of 75% in the identification of the root cause CI. If more than one cause is possible, you can investigate the most likely root cause before investigating less likely root causes.

By default, RCA groups with any confidence scores are listed. To limit what groups are listed, change the `sa_analytics.rca.query_probability_threshold` property to a percentage that the RCA group confidence score must meet to be listed. If a root cause CI has a confidence score that is lower than the specified percentage, alert aggregation and RCA does not treat that CI as the root cause.

View root cause CIs

You have these options for viewing root cause CIs in the Event Management dashboard.

<table>
<thead>
<tr>
<th>UI access</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>In a business service map</td>
<td>Displays root cause CIs highlighted in a business service map, and the relationships between the root cause CI, alerts, and related CIs in business services.</td>
</tr>
<tr>
<td>In impacted services</td>
<td>Displays all automated alert groups that are associated with business services. You can drill down to view details about the alerts in the group, and the root cause CI if it exists. Double-click a group, and then click the Impacted Services tab to display the services and root cause CIs if applicable.</td>
</tr>
</tbody>
</table>

RCA configurations

Event Management alert aggregation and RCA uses an RCA configuration to determine which alerts to include in its root cause analysis. The RCA configuration is used in learning the conditional probability of how a particular state of CI impacts other CIs. Multiple RCA configurations can be defined, but only a single configuration is in effect at any point of time for a given domain.

There are two types of RCA configurations, and the base system includes a pre-defined RCA configuration for each type:

- Rule-based model: Considers the severity of the alert, and maps any severity level (Critical, Major, Minor, Warning) into a state. For example, a binary-state configuration maps into two possible states:
  - 0 – No alert for the CI
  - 1 – Alerts exist for the CI

  Each configuration consists of one or more rules that define the set of alerts to be included in the RCA, such as all alerts with the severity of critical. You can define custom RCA configuration using the rule-based model, adding rules to map alerts according to CI and alert attributes to various states.

  The base system includes the predefined RCA configuration Default Binary Model Config, which is the default RCA configuration.

- Multi-state model: Based on the combination of the Resource and the Metric Name alert columns to learn the model, and it is not associated with any rules. The multi-state model combines the Metric Name and Resource alert columns into a string, and then aggregates
such strings from multiple alerts into a single state string. The number of resulting states is determined by the number of unique state strings for a particular CI.

The base system includes the predefined RCA configuration Default Multi State Config, which you can use in a comparison test of RCA configurations. There are no variations for this default configuration, and therefore you cannot create a custom RCA configuration that is based on the multi-state model.

There is no single RCA configuration that is the most optimal in every environment. Therefore, in addition to the default configurations, you can create custom rule-based RCA configuration. A custom RCA configuration can use additional CI and alert attributes, such as a location attribute to scope the analysis to a specific data center. User can add rules that define how to map alerts for different CIs to various states.

You can compare the RCA results on actual business data, between any two RCA configurations allowing you to select the optimal configuration in your environment. A comparison simulates analysis on historical data, and the RCA Configs Comparison report displays the results of the comparison.

Create an RCA configuration
Event Management alert aggregation and RCA uses an RCA configuration to determine which set of alerts from the sa_analytics_alert table to use for RCA. You can create a custom rule-based RCA configuration, compare it with another RCA configuration and decide how efficient and helpful it is. You can then choose the RCA configuration that is most suitable in your environment, and configure RCA to use that configuration in its analysis.

Role required: evt_mgmt_admin
Each rule-based RCA configuration consists of one or more rules. If an RCA configuration has multiple rules, then the rules’ filters are applied to alerts according to their priority order.

If the Domain Support - Domain Extensions Installer plugin is activated, then alerts are processed according to the domain of the associated business service.

1. Navigate to **Event Management > Alert Aggregation and RCA > RCA Configs**
2. Click **New**, and fill out the SA RCA SMC Config Base form.

### RCA Configuration form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Configuration name.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the configuration.</td>
</tr>
<tr>
<td>Domain</td>
<td>Domain that this configuration applies to.</td>
</tr>
<tr>
<td>Config deployed</td>
<td>Read-only field, which is automatically computed by the system. It is selected (true) if at least one version of the configuration has been deployed.</td>
</tr>
<tr>
<td>Created</td>
<td>Date and time that this configuration was created.</td>
</tr>
<tr>
<td>Created by</td>
<td>User that created this configuration.</td>
</tr>
<tr>
<td>Updated</td>
<td>Date and time that this configuration was updated.</td>
</tr>
<tr>
<td>Updated by</td>
<td>User that updated this configuration.</td>
</tr>
</tbody>
</table>
3. Right-click on the form title, and click **Save**.

4. Add rules to a rule-based RCA configuration:
   a) Click **SA RCA SMC Rule Bases** at the bottom of the form.
   b) Click **New**, and fill out the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the RCA configuration rule.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the RCA configuration rule.</td>
</tr>
<tr>
<td>Configuration id</td>
<td>The RCA configuration that this rule is associated with.</td>
</tr>
<tr>
<td>Order</td>
<td>An integer that represents the priority of the rule within the rest of the rules for the configuration. Lower numbers have higher priority.</td>
</tr>
<tr>
<td>Created by</td>
<td>User name.</td>
</tr>
<tr>
<td>Created</td>
<td>Time created.</td>
</tr>
<tr>
<td>Updated by</td>
<td>User name.</td>
</tr>
<tr>
<td>Updated</td>
<td>Time updated.</td>
</tr>
<tr>
<td>Domain</td>
<td>Domain that this rule applies to.</td>
</tr>
<tr>
<td>% Alert Coverage</td>
<td>System generated percentage of alerts in the sa_analytics_alert table that match the rule's filter.</td>
</tr>
<tr>
<td>Rule filter</td>
<td>Conditions to filter alerts for this rule.</td>
</tr>
</tbody>
</table>

5. Select the check box for the new rule and click **Alert Coverage** to calculate the rule's % Alert Coverage.

- Compare the new RCA configuration with another configuration, and decide which configuration is most helpful and should be deployed in the next RCA Learner cycle.
- After modifying a configuration, or when there is a significant number of new alerts, select any rules and click **Alert Coverage**. This recalculates the % Alert Coverage for the selected rules.

Compare and deploy RCA configurations

Compare the results of root cause analysis between two RCA configurations on actual alerts, then choose the optimal configuration in your environment, and deploy it.

Select any two RCA configurations and a business or an application service for the comparison. The comparison simulates analysis on historical data from the business or application service,
while applying the two RCA configurations. The RCA Config Comparison report displays the comparison results.

When you compare one RCA configuration to another RCA configuration, alert aggregation and RCA stores a current snapshot of the definitions of the RCA configurations that are being compared. An RCA configuration can end up having several versions, each one created when it is used in a comparison.

Review the comparison results, and decide which RCA configuration is most efficient, and then deploy that configuration. The deployment request is registered, and will take effect at the next cycle of the RCA Learner. The version of the configuration that was used for the comparison will be deployed.

1. Navigate to Event Management > Alert Aggregation and RCA > RCA Configs.
2. Click on one of the two configurations that you want to compare, and on the configuration form click Run Comparison.
3. In the Run Comparison dialog box, select a second configuration for the comparison from the Compare with Configuration list.
4. Select a business or an application service from the Run comparison on list to use for the comparison.
5. Click Run.
6. Navigate to Event Management > Alert Aggregation and RCA > RCA Configs Comparison, and select a comparison to review.
   Ensure that the status of the comparison is Completed.
7. Decide which configuration you want to deploy, and click the configuration link in the comparison that you are reviewing.
8. On the configuration form, click Deploy.

View root cause CIs in a business service map
For alerts associated with discovered business services or application services, you can view root cause CIs in a business service map on the Event Management dashboard.

Role required: evt_mgmt_admin, evt_mgmt_operator, or evt_mgmt_user

When you select a root cause CI, the business service map displays the business or application services that are impacted by the root cause CI.

**Note:** Session timeout settings do not apply to this screen, so the session will remain connected, even when there is no human interaction. If this is a concern, either log out or close the active tab in the browser.

1. Navigate to Event Management > Dashboard.
2. Double-click the tile of the business or application service for which you want to view root cause CIs.
3. Underneath the map, click Root Cause CI and select a root cause CI to highlight in the business service map.

The list of root cause CIs associated with the selected business service displays the following details:

<table>
<thead>
<tr>
<th>Root Cause CI details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Field</strong></td>
</tr>
<tr>
<td>Root Cause CI</td>
</tr>
<tr>
<td>Related Alerts</td>
</tr>
</tbody>
</table>
4. Double-click a root cause CI to display the alerts within this alert group.
5. Select any alert to highlight the CIs that it is associated with, and view additional details about the alert.

**Alert aggregation and RCA reports**

Event Management alert aggregation and RCA generates several reports to help you assess the efficiency and depth of analysis.

You can access alert aggregation and RCA reports by navigating to **Event Management > Alert Aggregation and RCA**.

**RCA Config Comparison Report**

This report displays the results of a comparison between two RCA configurations defined for root cause analysis. Evaluate and score the comparison results. Then choose the RCA configuration that appears most efficient in your environment, and deploy it.

In addition to the default RCA configurations, you can create custom rule-based RCA configurations, and you can compare any two configurations (ConfigA and ConfigB). For each configuration that is being compared, RCA algorithms are applied using real historical alerts from the past 7 days. The RCA Config Comparison Report displays the RCA results for each configuration, if it would have been used during the past 7 days. Once you start a comparison, wait for the report to show that the status is **Completed**.

To display the report, navigate to **Event Management > Alert Aggregation and RCA > RCA Configs Comparison**, which requires the evt_mgmt_admin role. The report has 3 sections, each section provides further details for a selected item in the previous section.

**Comparison summary list**

This section lists summaries of recent comparisons, with the following details:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start time</td>
<td>Time the comparison started.</td>
</tr>
<tr>
<td>Run Name</td>
<td>System created unique ID for the comparison run, prefixed by ‘RUN’.</td>
</tr>
<tr>
<td>Status</td>
<td>Status of the comparison, ensure it is completed before reviewing the details.</td>
</tr>
<tr>
<td>Config (A)</td>
<td>The first RCA configuration used in the comparison.</td>
</tr>
<tr>
<td>Config (B)</td>
<td>The second RCA configuration used in the comparison.</td>
</tr>
<tr>
<td>Score (A)</td>
<td>Sum of user assigned scores for all automated alert groups analyzed by Config (A).</td>
</tr>
<tr>
<td>Score (B)</td>
<td>Sum of user assigned scores for all automated alert groups analyzed by Config (B).</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Service</td>
<td>Discovered business service or application service for which alerts were used in this comparison.</td>
</tr>
</tbody>
</table>

Actions:
- Select a comparison for which to display further details about RCA results for Config (A) alongside Config (B).
- Deploy an RCA configuration:
  1. Click the link for the RCA configuration that you want to deploy in the **Config (A)** or **Config (B)** column.
  2. On the configuration form, click **Deploy**.

**RCA results for Config (A) and Config (B) per comparison**

This section displays a comparison of root cause analysis for Config (A) and Config (B), for the selected comparison in the previous section. Displaying the following details:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grouping Time</td>
<td>Time that the group was created.</td>
</tr>
<tr>
<td>Root Cause</td>
<td>Comparison of the root cause CI being identified by Config (A) and Config (B). Both configurations have either identified the <strong>Same</strong> root cause CI, or a <strong>Different</strong> one.</td>
</tr>
<tr>
<td>Score (A)</td>
<td>User assigned score of RCA results for this automated alert group, as it was analyzed by Config (A).</td>
</tr>
<tr>
<td>Score (B)</td>
<td>User assigned score of RCA results for this automated alert group, as it was analyzed by Config (B).</td>
</tr>
<tr>
<td>Probability % (A)</td>
<td>Confidence in the accuracy of the root cause CI identified by Config (A).</td>
</tr>
<tr>
<td>Probability % (B)</td>
<td>Confidence in the accuracy of the root cause CI identified by Config (B).</td>
</tr>
<tr>
<td>Group Name (A)</td>
<td>Automated alert group created by Config (A).</td>
</tr>
<tr>
<td>Group Name (B)</td>
<td>Automated alert group created by Config (B).</td>
</tr>
<tr>
<td>Related Alerts (A)</td>
<td>Summary of alerts within the automated alert group created by Config (A), broken down by alert severity.</td>
</tr>
<tr>
<td>Related Alerts (B)</td>
<td>Summary of alerts within the automated alert group created by Config (B), broken down by alert severity.</td>
</tr>
</tbody>
</table>

Actions:
- Select an automated alert group to display further details about all the alerts within that group.
- Score results:
1. Select the group and the configuration that you want to score, in either *Score (A)* or *Score (B)* column.

2. Double-click the score value, enter a numeric score, and then click the green check-mark. The new group score value is aggregated into the summary score for *Score (A)* or *Score (B)* in the comparison summary row for this comparison.

- Toggle between different display options for the automated alert groups:
  - **Same**: Displays only the automated alert groups for which RCA results for Config (A) and Config (B) are identical.
  - **Different**: Displays only the automated alert groups for which RCA results for Config (A) and Config (B) are different.
  - **All**: Displays all automated alert groups for which there are RCA results for either Config (A) or Config (B).

**Correlated alert group details per config**

This section displays details of all related alerts within an automated alert group selected in the previous section. Displaying the following details for both Config (A) and Config (B):

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>Alert ID.</td>
</tr>
<tr>
<td>Severity</td>
<td>Severity of the alert.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the alert.</td>
</tr>
<tr>
<td>Configuration Item</td>
<td>CI that the alert is associated with.</td>
</tr>
<tr>
<td>Updated On</td>
<td>Last time that the alert was updated.</td>
</tr>
</tbody>
</table>

**Learned Patterns Report**

This report displays metrics associated with learned patterns such as the frequency of occurrence and the number of occurrences of the pattern identifier attributes in learned patterns (CI/MetricName by default). You can sort the learned patterns by pattern score, pattern frequency, and pattern size. This report helps you identify higher-frequency occurring alert patterns.

For each alert, Event Management alert aggregation and RCA combines the associated pattern identifier attributes that represent the alert. It then groups all the pattern identifier attributes for alerts that occur together, to create a learned pattern. The same combination of pattern identifier attributes can exist in multiple learned patterns. Typically, a learned pattern indicates a single problem. The report displays metrics in table format, grouped by learned patterns.

To display the report, navigate to **Event Management > Alert Aggregation and RCA > Learned Patterns**.

You can use this report to:

- Allocate resources to the higher frequency occurring alerts to reduce the time for closing a large number of alerts.
- Allocate resources to larger learned patterns of lower frequency.

**Actions:**

- Expand a pattern group to display details about all the identifier attributes that are associated with the pattern.
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration Item</td>
<td>The configuration item used for the combined pattern identifier attributes.</td>
</tr>
<tr>
<td>Feature Identifier</td>
<td>The feature identifier used for the combined pattern identifier attributes.</td>
</tr>
<tr>
<td>Frequency</td>
<td>The number of times that the combined pattern identifier attributes occur in the data set.</td>
</tr>
</tbody>
</table>

- Sort within each pattern by Configuration Item and Feature Identifier.
- View the following information for each pattern:
  - **Size**: The number of combined pattern identifier attributes occurrences in a learned pattern.
  - **Frequency**: The number of times that a learned pattern occurs in the data set.
  - **Score**: Combined calculation of frequency and size, using the formula:
    \[ \text{Frequency} \times (\text{Size} + 1) \]
- Click on a CI link in the Configuration Item column to display the CI form.

**Event Management Operator Tutorial**

As an Event Management operator, your role is to find alerts, analyze them, and take action to help resolve the underlying issue.

**Take the tutorial**

Before you can work as an operator, you should take a tutorial to become familiar with several Event Management concepts and understand what operators typically do on a regular basis.

Start the tutorial by clicking the first lesson:

<table>
<thead>
<tr>
<th>Lesson</th>
<th>Lesson title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lesson 1</td>
<td>An overview of events and alerts</td>
</tr>
<tr>
<td>Lesson 2</td>
<td>An overview of business services</td>
</tr>
<tr>
<td>Lesson 3</td>
<td>Event management operator workspaces</td>
</tr>
<tr>
<td>Lesson 4</td>
<td>What operators do</td>
</tr>
</tbody>
</table>

**An overview of alerts for Event Management operators**

As an Event Management operator, you need to understand how an alert is generated from an event, what to look for in an alert, and how alerts can be grouped together.

This is the first lesson in the Event Management tutorial.

<table>
<thead>
<tr>
<th>Lesson</th>
<th>Lesson title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lesson 1</td>
<td>An overview of events and alerts</td>
</tr>
<tr>
<td>Lesson 2</td>
<td>An overview of business services</td>
</tr>
</tbody>
</table>
Your organization already has an event monitoring tool in place, such as Microsoft System Center Operations Manager (SCOM), Nagios, SolarWinds, and so on. When an issue occurs on your network, such as a computer going down or a database failure, these event monitoring tools send events to your ServiceNow instance. The Event Management application processes these events according to the settings that your administrator configured, and then it generates alerts. An alert is an indicator that the issue requires some type of action.

That is where you come in as an Event Management operator. Your role is to view alerts and, depending on how Event Management is implemented in your organization, take some sort of action to help resolve the underlying issue or notify someone else who can. Later in this tutorial, you can see the phases of a typical alert management process.
Alert priority and severity

The two most common characteristics of an alert are the priority and the severity.

- The **priority** of an alert is a score that helps you determine how important the impact is to business services. Multiple factors determine the alert priority score. Your Event Management administrator can configure the algorithm that Event Management uses to calculate priority.
- The **severity** of an alert is an indicator of how serious the underlying issue is. The event monitoring tool in your organization usually sends severity values as part of the event, which then gets carried over as part of the alert. These are the default severity types that you can see in this tutorial:

<table>
<thead>
<tr>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The resource is either not functional or critical problems are imminent.</td>
</tr>
<tr>
<td>Major</td>
<td>Major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td>Minor</td>
<td>Partial, non-critical loss of functionality or performance degradation occurred.</td>
</tr>
<tr>
<td>Warning</td>
<td>Attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td>Info</td>
<td>No severity. An alert is created. The resource is still functional.</td>
</tr>
<tr>
<td>Clear</td>
<td>The alert no longer needs action.</td>
</tr>
</tbody>
</table>

Correlated alerts

Some alerts are related to each other. For example, if an issue occurs like a router going down, several separate alerts could be generated, one for each server connected to the router. All of these alerts are actually related, or **correlated**. To help you manage these correlated alerts, Event Management can automatically group them together and establish a two-level hierarchy with one root alert, called the **primary alert**, at the top, and other related alerts, called **secondary alerts**, under the primary alert. When you view alerts, primary alerts stand out by default so you know which alert to focus on without being distracted by the secondary alerts.

In our example, if a router goes down on your network, network communication is also affected for connected servers, assuming they cannot reach any other routers. The router outage becomes the primary alert and the alerts generated on the server are secondary alerts that should be correlated under the router alert.
Depending on your organization’s Event Management implementation, alerts might be grouped automatically based on correlation rules that your administrator sets up. Your ServiceNow instance can also learn how to improve the way it automatically correlates alerts based on these rules and feedback that you can give. As an operator, you should still verify the accuracy of the correlation and, if necessary, manually correlate additional alerts with the primary alert. Later in the tutorial, you will learn how to do this.

In this tutorial, you will learn how to manually correlate alerts. In an advanced topic, you will learn how to give feedback to the system, so that your system can improve the automatic alert correlation process.

**Alert flapping**

An alert can flap, meaning that it gets multiple open-close events in rapid succession. Flapping indicates that Event Management does not know if the underlying events are genuine or not. The events could indicate small issues with the way CIs are configured, or larger issues, like network outages.
For example, if a server that hosts a web service has too many active processes, it might trigger an event about excessive CPU usage. Since CPU usage can fluctuate rapidly depending on the requests made through the web service, several events might be triggered, leading to the alert being put in the flapping state. As an operator, you might need to create an incident to have the server restarted, or someone might have to reconfigure the CPU, or possibly make a hardware change on the device.

As another example, consider a loose network cable that causes momentary, repeated network outages. The thresholds that your administrator configures might not be optimal for this kind of alert and Event Management considers it a flapping alert.

Continue the tutorial

Proceed to the next lesson: Business services for Event Management operators.
Business services for Event Management operators

As an Event Management operator, you need to understand what business services are. This is the second lesson in the Event Management tutorial.

| Lesson 1 | ✓ | An overview of events and alerts |
| Lesson 2 | ☐ | An overview of business services |
| Lesson 3 | ☐ | Event management operator workspaces |
| Lesson 4 | ☐ | What operators do |

A business service is a collection of components, such as network devices, computers, and applications, that offer a service to your organization. These services can be something like an email system or a website that tracks orders or requests in a database. Your administrator should have already specified the business services in your ServiceNow instance.

Each component that makes up a business service is referred to as a configuration item or CI. This is a visual representation of a business service on a business service map:
In this example, you can see a Web Portal business service with these CIs:

- **PS Apache03**: An Apache web server that hosts a company website.
- **PS LinuxApp01** and **PS LinuxApp02**: Two Linux servers that share the workload from the web server.
- **PS ORA01**: A database server that both Linux servers need to access.
- **Storage Area Network 001**: A mass storage device on which the other CIs depend.

You can see business service maps like this from the Event Management Dashboard. Later in this tutorial, you will learn about the dashboard and what business services look like when an alert is associated with a CI.

**Types of business services**

Event Management classifies business services into the following types:

**Technical services**
A technical service is a dynamic grouping of CIs, based on some common criteria. For example, a technical service could be comprised of all web servers or all Oracle databases for a specific location, like North America.

**Application services**

An application service is an ad-hoc business service that your administrator puts together by selecting each CI separately and choosing the relationship between them.

**Discovered services**

A discovered service is business service that [Service Mapping](#) finds (if your organization uses it).

**Alert groups**

Alerts that are grouped together, either manually or automatically.

**Alert impact**

Business services are critical to the operations of your organization, so if an issue occurs on one CI, the entire business service can be impacted. Part of your role as an operator is to analyze alerts on CIs and see how they impact the business service as a whole, and then take some sort of action to help remediate or solve the underlying issue. Your administrator can configure impact rules that go into calculations for the severity of an alert.

Later on, when you learn how to use the Event Management Dashboard, you will learn how to view an impact tree for a business service so you can understand the relationship between the severity of an alert and the overall business service.

In this example, you can see how a **Major** alert on an Oracle database (PS ORA01) also causes a major alert on its parent CIs and on the Web portal business service itself.
Continue the tutorial

Proceed to the next lesson: Event Management operator environment.

Event Management operator environment

As an Event Management operator, your primary work environment is the Event Management dashboard and Alert Intelligence.

In this, the third lesson in the Event Management tutorial, you are given an overview of the workspaces in which you work, so you know how to find the information you need.

<table>
<thead>
<tr>
<th>Tutorial lessons</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Lesson 1</td>
<td><em>An overview of events and alerts</em></td>
</tr>
<tr>
<td>Lesson 2</td>
<td><em>An overview of business services</em></td>
</tr>
<tr>
<td>Lesson 3</td>
<td>Event Management operator environment</td>
</tr>
<tr>
<td>Lesson 4</td>
<td><em>What operators do</em></td>
</tr>
</tbody>
</table>
The Event Management dashboard

Your main area of work is the Event Management dashboard, which provides an application-service centric view of all alerts. From here, you can drill into each application service to see the affected CIs and get an understanding of the overall impact of whatever caused the alert.

The main sections of the dashboard are:
The banner contains controls that show or hide application services based on the criteria you select:

- Use the application service selector to choose a specific application service to display in the main body of the dashboard.

- Use the severity slider to show or hide application services based on the alerts associated with them.

- From the Prioritize by choice list, order the application services by Severity, Criticality, or Cost.

  **Note:** Your administrator assigns an application criticality and cost value to your application services. You do not have to configure these.

- Select Groups to show the application service groups that your administrator set up or select Services to show application services separately.

**Application services**

On the main part of the Event Management dashboard are tiles, each of which is an application service. Each tile has a color, which represents the severity of the alerts that are associated with the application service, and a size, which represents the priority of the alerts.
Click an application service tile to show only the alerts associated with that application service. A gray box outlines the tile and the alerts appear in the Alerts list at the bottom.

### Alerts

All active alerts appear in the Alerts list under the dashboard. You can also filter or sort the list to find the alert you are looking for:

Sort by any alert details, such as the **Priority**, which considers multiple factors for how serious the alert is, or the **Severity**, which is value provided by the event monitoring tool.
Click the **Correlated Alerts** switch to the off position to show all alerts, regardless of whether they are secondary alerts that are already correlated with a primary alert. By default, the switch is in the on position, grouping correlated alerts under the primary alert.

Open any alert by clicking the number.

You will learn about what each of the columns means for an alert later on when you analyze an alert.

**Application service map views on the Event Management Dashboard**

Double-click the name of an application service tile to open one of the application service map views. The view that you see depends on the type of application service.

*For a manual or standard application service, this view appears:*
The application service map that you were introduced to in a previous lesson appears in the main panel. This is what you can do from this view:

Click any of the CIs to see the alerts only for that CI and to display the details about that CI in the Properties pane.
Click **Impact Tree** to see the state of all the CIs and how they affect each other when receiving an alert. You will learn more about the impact tree later on in the tutorial.

Open any alert from the **Alerts** list at the bottom by clicking the number.
For a technical application service, this view appears:

This is what you can do from this view:
Click any CI to view details about it.

<table>
<thead>
<tr>
<th>Name</th>
<th>Severity</th>
<th>Class</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>PS ORA01</td>
<td>Major</td>
<td>Database</td>
<td></td>
</tr>
<tr>
<td>lawson_db</td>
<td>Critical</td>
<td>Database</td>
<td>lawson_db_100</td>
</tr>
<tr>
<td>MySQL-16011</td>
<td></td>
<td>Database</td>
<td></td>
</tr>
<tr>
<td>bond_trade_ny</td>
<td></td>
<td>Database</td>
<td></td>
</tr>
<tr>
<td>SAP ORA01</td>
<td></td>
<td>Database</td>
<td></td>
</tr>
<tr>
<td>bond_trade_aus</td>
<td></td>
<td>Database</td>
<td></td>
</tr>
</tbody>
</table>

Open any alert from the Alerts list at the bottom by clicking the number.

Alert intelligence

In the navigation pane, navigate to Event Management > Alert Intelligence.

Alert intelligence provides insight data analysis on the alert and related CIs based on history, analysis of CMDB, and ITSM data. You can, for example, see repeated or similar alerts and this information can help you to investigate problems further.

For more information, see View alert insight information in Workspace.

Continue the tutorial

Proceed to the next lesson: What Event Management operators do.
What Event Management operators do

As an Event Management operator, your typical workflow involves three phases: analyzing an alert and its effect on business services, taking some type of action, and making sure the alert is finally closed.

This is the last lesson in the Event Management tutorial. In this lesson, you will run through the three phases of a typical operator day-to-day workflow.

| Lesson 1 | An overview of events and alerts |
| Lesson 2 | An overview of business services |
| Lesson 3 | Event management operator workspaces |
| Lesson 4 | What operators do |

Your organization will have specific policies and procedures that may differ from the phases outlined in this topic. In addition, your administrator might have customized the Event Management application so that some of these phases are automated and do not actually require any action on your part.

For the purposes of this tutorial, we will walk you through each phase and allow you to perform the tasks manually.

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Analyze and acknowledge an alert</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>In this phase, you will find an alert to work on, analyze the details, and acknowledge it so that other operators know it is a legitimate alert.</td>
</tr>
<tr>
<td>Phase 2</td>
<td>Triage an alert</td>
</tr>
<tr>
<td></td>
<td>In this phase, you take an action to help remediate the issue that caused the alert. The most common action to take is to create an incident and assign it to someone who can solve the underlying issue.</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Close an alert</td>
</tr>
<tr>
<td></td>
<td>In this phase, you will verify that the alert is resolved, and then close the alert.</td>
</tr>
</tbody>
</table>

Start here

Start out by learning how to *Analyze and acknowledge an alert*.

*Operator phase 1: Analyze and acknowledge an alert*

As an Event Management operator, the first thing you should do is access alerts and then find the ones you want to focus on. You can open the Alert form to analyze the details, and then acknowledge it to let other operators know that the issue causing the alert should be addressed in some way.
You must understand both of your operator environments: the Event Management Dashboard and Alert Intelligence. Review Event Management operator environment if you have not already done so. This tutorial uses the Event Management dashboard.

1. Navigate to Event Management > Dashboard.
   All the alerts are listed at the bottom of the dashboard under the business services.
2. Find the alerts you want to focus on by clicking a business service, or by using any of the filtering and sorting options in the banner.
3. Click the alert number to open it.
   In this example, we are focusing on the Web Portal business service and the alert with the highest severity.

4. Review the important fields and related lists on the form. These are described below:
The Alert form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>This is the automatically generated alert number that you saw in the list of alerts on the Event Management Dashboard or the Alert console.</td>
</tr>
<tr>
<td>Node</td>
<td>The name of the CI with which the alert is associated. The event that generated the alert occurred on this CI. You might see a computer name, or an IP address or MAC address of a device.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>The object on the node that caused the event. You might see values like the name of a hard disk, a CPU, or the name of a process or a service. You can add information to this field if necessary.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>The CI in your database to which this alert applies. If the Node field is filled in but this field is empty, and if you know which CI the node represents, you can enter the CI in this field to manually associate the alert with it.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Your administrator can configure rules that automatically bind CIs to an alert, usually based on what is in the Node field. However, it is possible that the system is unable to find the correct CI to bind the alert to. In this case, you might need to manually select a CI. For the purposes of this tutorial, we assume that the correct CI is bound to the alert.</td>
</tr>
<tr>
<td>Task</td>
<td>Any existing task that is associated with the alert, such as an incident, change request, or problem, is shown in this field. When you first look at an alert, this field is blank. You can associate a task when you triage the alert. You associate an incident to an alert and assign the incident to someone at a later stage in this tutorial.</td>
</tr>
<tr>
<td>Description</td>
<td>A description of the alert that came from the alert monitoring software in your organization. You can add information if necessary.</td>
</tr>
<tr>
<td>Severity</td>
<td>The severity value that is copied from the event:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Critical</strong> indicates that immediate action is required. The CI is not functional or critical problems are imminent.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Major</strong> indicates that major functionality is severely impaired or performance has degraded.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Minor</strong> indicates that there is a partial, non-critical loss of functionality, or that performance had degraded.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Warning</strong> indicates that attention is required, even though the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Info</strong> Indicates some type of informational message, but the resource is still functional.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Clear</strong> indicates that no action is required.</td>
</tr>
<tr>
<td>State</td>
<td>Indicates the processing state that the alert is in. A newly generated alert is in the <strong>Open</strong> state. Other states are:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Reopen</strong> indicates that a previously closed alert is open again, and it requires your attention.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Flapping</strong> indicates that the alert is receiving a high frequency of identical events from the same source, which can cause many alerts to re-open even after it is closed.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Closed</strong> indicates that the alert is closed and does not require any further action. You will close an alert as part of the last stage in this tutorial.</td>
</tr>
</tbody>
</table>
### Field Description

- **Acknowledged**: Indicates that an operator verified that the alert is genuine and is triaging it. You acknowledge the alert at a later stage in this topic.
  
  Your Event Management administrator can also have alerts automatically acknowledged immediately after they are generated. If this is the case, the **Acknowledged** check box on the Alert form might already be selected.

- **Maintenance**: Indicates that the alert does not need to be worked on and is hidden from the list of alerts in the Event Management Dashboard and the Alert Console. In this example, we do not want to put the alert in maintenance.

- **Parent**: Displays a value only for secondary alerts. The value indicates the alert number of the parent alert. The field is empty for primary alerts (alerts with no parent alert).

- **Knowledge article**: Any existing knowledge base (KB) article that is associated with the alert. When you first open an alert, this field is blank. You can associate a KB article when you triage the alert.

### Instructions

5. **Navigate to back to** **Event Management > Dashboard**.
6. **Double-click the business service tile for the Web portal**.

In this example, the business service is an application service, not a technical service, so the business service map appears.
Notice that the major severity indicator in orange is displayed on several CIs. It starts with the Oracle database (PS ORA01) and propagates up to all CIs that depend on the database. From this, you can surmise that the alert with the highest level of severity on the Oracle database indicates the root cause of the issue.

This does not necessarily mean that the system generated separate alerts for each CI, but it is possible. When you triage an alert, you can see if alerts on other CIs should be correlated under a primary alert.

7. Click the **Impact Tree** tab and review the impacts on each CI.

The percentages next to each CI in the impact tree show you the affect on each CI. In this example, each CI is impacted 100% by the alert, which means the underlying issue that led to the alert is serious enough to impact the functionality of the business service.
8. Open the alert on the Oracle database that you first opened above.
9. Click the **Acknowledge** button at the top of the form.

Acknowledging the alert does not assign it to you, nor does it create a task like an incident or change request. It simply lets other operators know that you are aware of the issue. After you acknowledge it, you will take further action during the triage stage.

10. Navigate back to the Event Management Dashboard or the Alert Console, and look for the alert that you acknowledged. You can see that the **Acknowledged** value is true.

Go to **Operator phase 2: Triage an alert**.

**Operator phase 2: Triage an alert**

After you analyze and acknowledge an alert, you must triage it. The triage phase involves verifying alert correlation and taking some sort of action to help resolve the issue that caused the alert. This topic covers the most common triage task: creating an incident from an alert.

---

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
1. Before you start to work on the alert, check if other alerts need to be correlated with the one you just reviewed:
   a) Navigate to **Event Management > Dashboard**.
   b) Click the same business service tile and review the alerts in the **Alerts** list.
   c) If multiple alerts are present, open one to determine if it is related to the other alert, or if the underlying issue is the same in both alerts. For example, if an alert is associated with a web server and there is already an alert for the database issue, correlate the web server alert as a **secondary** alert with the database alert as the **primary** alert.
   d) Open the secondary alert, click the lookup icon (🔍) in the **Parent** field, and then select the primary alert. This creates a primary (parent) / secondary (child) relationship between the two alerts.
   e) Repeat this process to correlate as many secondary alerts as necessary.
   f) Navigate back to the Event Management Dashboard and verify that the primary alert displays an icon in the **Group** column.
2. Open the primary alert.
3. On the Alert form, click Create Incident. If your organization uses Security Incident Response, the button is named Create Security incident.

4. Fill in the form fields (see table below):

Note: Your administrator can set up rules that lets the system automatically correlate alerts so that you do not need to do so manually. In that case, a Feedback field appears on the Alert form and you can select Yes if the system correlated the alert correlated correctly, or No if it was not.
Incident INC0010004

- **Number**: INC0010004
- **Caller**: EM Operator
- **Category**: Database
- **Subcategory**: -- None --
- **Business service**: Web portal
- **Configuration item**: PS ORA01

**Short description**: Database issue

**Description**: Database issue

**Watch list**: [Lock] [Unlock]

**Work notes**: The Oracle Database service should be restarted or checked for other issues.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caller</td>
<td>Click the lookup icon and then select your name. The caller is the person who discovered the issue that led to the incident.</td>
</tr>
<tr>
<td>Category and Subcategory</td>
<td>Select a category that best describes this alert. In this example, Database is the best choice.</td>
</tr>
<tr>
<td>Business service</td>
<td>Select the business service to which the CI belongs. In this example, the business service is Web portal.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>If it is not already populated automatically, select the CI. In this example, the Oracle database (PS ORA01) is automatically populated into the field.</td>
</tr>
<tr>
<td>Impact and Urgency</td>
<td>Select the impact and urgency levels that you think is appropriate.</td>
</tr>
<tr>
<td>Assignment group Assigned to</td>
<td>Click the lookup icons ( ) for both or either of these fields, and then select the group or the individual that can handle the issue.</td>
</tr>
</tbody>
</table>

5. Click **Submit** on the Incident form to create the incident. The Alert you are working on reappears.

On the Alert form, the incident is populated in the **Task** field. You can also see the incident number in the Task column on the Event Management Dashboard.

There are other tasks you perform as part of the triage stage:

- **Run a remediation workflow on an alert** if your Event Management administrator already set up a workflow in your ServiceNow instance and your policies allow you to trigger it from the alert.
- **Launch a web application from an alert** to open a website or an event monitoring tool that provides more information about the alert.
- **Put an alert into maintenance** to temporarily hide it from the Alert Console and Event Management Dashboard if the alert does not require action at this time.
- **Associate a knowledge base article with an alert** if there is existing information about the alert that might help resolve the underlying issue.

If you do not need to perform any other triage actions, proceed to **Phase 3: Close an alert**.

Run a remediation workflow on an alert

As an Event Management operator, you can also run a workflow on your ServiceNow instance that helps remediate the alert. For example, you might run a workflow that automatically restarts a server on your network, which might resolve an alert about CPU usage.

Role required: evt_mgmt_operator

Phase 1: Analyze and acknowledge an alert
### Phase 2
- **Triage alerts**

### Phase 3
- **Close an alert**

**Note:** You can run a remediation workflow if your administrator already set up workflows for you to choose from. You should be familiar with your organization’s policies regarding triaging of alerts.

1. From the alert console or the Event Management Dashboard, open the alert that you acknowledged in Phase 1: Analyze and acknowledge an alert.
2. On the Alert form, click **Quick Response**.
3. In the Quick Response window, click the name of the remediation under **Run Remediation**.

There are also other tasks you can take as part of the triage stage:
- **Launch a web application from an alert** to open a website or an event monitoring tool that provides more information about the alert.
- **Put an alert into maintenance** to temporarily hide it from the Alert Console and Event Management Dashboard if the alert does not require action at this time.
- **Associate a knowledge base article with an alert** if there is existing information about the alert that might help resolve the underlying issue.

If you do not need to perform any other triage actions, proceed to **Phase 3: Close an alert**.

Launch a web application from an alert

As an Event Management operator, you can also launch a web application from an alert. The web application could be a console for the event monitoring tool that your organization uses, or any external website that provides additional information you might need about the alert.

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Analyze and acknowledge an alert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 2</td>
<td>Triage alerts</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Close an alert</td>
</tr>
</tbody>
</table>

You can launch a web application if your administrator already set up a link to the application. You should be familiar with your organization’s policies regarding triaging of alerts.

1. From the alert console or the Event Management Dashboard, open the alert that you acknowledged in Phase 1: Analyze and acknowledge an alert.
2. On the Alert form, click **Quick Response**.
3. In the Quick Response window, click the name of the application under **Launch Application**.
There are also other tasks you can take as part of the triage stage:

- **Run a remediation workflow on an alert** if your Event Management administrator already set up a workflow in your ServiceNow instance and your policies allow you to trigger it from the alert.
- **Put an alert into maintenance** to temporarily hide it from the Alert Console and Event Management Dashboard if the alert does not require action at this time.
- **Associate a knowledge base article with an alert** if there is existing information about the alert that might help resolve the underlying issue.

If you do not need to perform any other triage actions, proceed to **Phase 3: Close an alert**.

**Associate a knowledge base article with an alert**

As an Event Management operator, you can associate a knowledge base (KB) article with the alert to capture additional information about the alert. This might include a procedure that someone has to follow to resolve the underlying issue on your network, or a best practice to prevent the issue from reoccurring.

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Analyze and acknowledge an alert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 2</td>
<td>Triage alerts</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Close an alert</td>
</tr>
</tbody>
</table>

This task assumes that your organization uses the Knowledge Base application in your ServiceNow instance.

1. From the alert console or the Event Management Dashboard, open the alert that you acknowledged in Phase 1: Analyze and acknowledge an alert.
2. On the Alert form, click the lookup icon (🔍) next to the **Knowledge article** field.

3. Filter the list of existing KB articles by first selecting a field, such as **Short Description**, and then entering related text into the search text field. You can use the `contains (*)` operator to search for articles that contain keywords. For example, entering `*oracle` in the short description filters the KB articles that contain the word `oracle` somewhere in the short description.

4. If you cannot find any related KB articles, you can click **New**, create a new one, and then click **Submit**.

The KB article number appears in the **Knowledge article** field on the Alert form.

5. Click **Update** on the Alert form to save the information.

There are also other tasks you can take as part of the triage stage:

- **Run a remediation workflow on an alert** if your Event Management administrator already set up a workflow in your ServiceNow instance and your policies allow you to trigger it from the alert.
- **Launch a web application from an alert** to open a website or an event monitoring tool that provides more information about the alert.
- **Put an alert into maintenance** to temporarily hide it from the Alert Console and Event Management Dashboard if the alert does not require action at this time.

If you do not need to perform any other triage actions, proceed to **Phase 3: Close an alert**.

Put an alert into maintenance
As an Event Management operator, you can put an alert into maintenance if the alert does not require any further action, but you still want to keep the alert active. Putting the alert into
maintenance hides it from the Alert Console and Event Management Dashboard so that other operators do not need to access it, but it does not close the alert.

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Analyze and acknowledge an alert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 2</td>
<td>Triage alerts</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Close an alert</td>
</tr>
</tbody>
</table>

1. From the alert console or the Event Management Dashboard, open the alert that you acknowledged in Phase 1: Analyze and acknowledge an alert.
2. Select the Maintenance check box and click Update, or click the Maintenance button at the top of the list.
3. Click Update.

Notice that alerts in maintenance are not visible in the list by default. The filter excludes alerts, where Maintenance | = | true.

If the issue that led to the alert still needs further attention later on, navigate to Event Management > All Alerts, open the alert, and then clear the Maintenance check box and click Update. From there you can then perform other triage actions.

There are also other tasks you can take as part of the triage stage:

- Run a remediation workflow on an alert if your Event Management administrator already set up a workflow in your ServiceNow instance and your policies allow you to trigger it from the alert.
- Launch a web application from an alert to open a website or an event monitoring tool that provides more information about the alert.
- Associate a knowledge base article with an alert if there is existing information about the alert that might help resolve the underlying issue.

If you do not need to perform any other triage actions, proceed to **Phase 3: Close an alert**.

**Operator phase 3: Close an alert**

After you take action on an alert, you can verify several items on an alert and then close it.

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Analyze and acknowledge an alert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 2</td>
<td>Triage alerts</td>
</tr>
<tr>
<td>Phase 3</td>
<td>Close an alert</td>
</tr>
</tbody>
</table>
1. Navigate to Event Management > Alert Console.
2. Find the alert you want to close. If the alert does not appear in the list for any reason, such as because it is in maintenance, click the filter icon and change the filtering options as needed.
3. Click the alert number in the Number column to open the Alert form.
4. Review key information about the alert in the main part of the Alert form:

   a) On the Task field, click the preview icon to see a small summary of the incident or task you created. You can also click Open Record to see the full task.
If the task is not assigned or is still in a Draft or Work in Progress state, you might not want to close the alert, per your organization’s policies. You can even select another task, such as another incident, a change request, or a problem to replace the existing one.

In this example, we assume everything is correct with the incident that you already created.

b) On the Knowledge article field, click the preview icon to see a small summary of a KB article and verify that captures the necessary information about the alert. You can also click Open Record to see the full article.

In this example, we assume the KB contains the necessary information.

5. Review information about business services in the Impacted Services related list. If you need more information about a business service and want to see the business service map, click View Service. In this example, the Web portal business service is visible.
6. Look at the alerts related to this one, and modify the correlation if necessary. This process is different if you are viewing a primary alert or secondary alert:

- On a primary alert, the secondary alerts that are correlated with the primary appear in the Alerts related list. In this example, the alert is a primary and the secondary alert that you already correlated with it during the triage stage appears:

Typically, you do not have to do anything with the secondary alerts, unless you want to add more secondary alerts, or remove the correlation altogether.

**Note:** When you close a primary alert, the secondary alerts are automatically closed by default.

- On a secondary alert:

  The alert number of the primary alert appears in the Parent field. You do not have to do anything with the parent alert, unless you want to re-correlate this secondary alert with another primary alert, or remove the correlation altogether by clearing the field.
When you are sure the underlying issue causing the generation of the event and corresponding alert has been rectified, click Close. If you think that the alert needs attention, put it into maintenance by clicking Maintenance instead of closing it.

You are finished with the Event Management Operator tutorial, which covers a basic workflow of analyzing alerts, triaging them, and closing them out.

After you close an alert, you might need to follow up on these items:

**If you need to reopen an alert**

You can also manually change the state of an alert to Reopen if you think it requires further action. When an alert reopens, any associated incidents can also be updated or reopened according to the incident state and the rules that your administrator configured. Always go through the triage and close phases again any time you open an alert.

**Alerts in the flapping state**

In some cases, the same event that created the alert you just closed could be triggered again, even repeatedly, in a short amount of time. If this occurs, the State of the alert you just closed changes to Reopen, and eventually to Flapping depending on settings that your administrator configures.

See *Handle alerts in the flapping state* for more information on what to look for.

Also see *Advanced tasks for the Event Management operator* for a summary of other tasks that you might need to perform at some time.

**Advanced tasks for the Event Management operator**

As an Event Management operator, you might need to perform additional tasks that are outside of your typical workflow, or tasks that you need to perform only once.

<table>
<thead>
<tr>
<th>Advanced task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Customize your Event Management Dashboard view</strong></td>
<td>Get to the alerts you need to analyze as soon as possible.</td>
</tr>
<tr>
<td><strong>Handle alerts in the flapping state</strong></td>
<td>If an alert is in the flapping state, it means that the underlying issue keep reoccurring. You should triage the alert again and look for another solution.</td>
</tr>
<tr>
<td><strong>Handle alerts while CIs are in maintenance</strong></td>
<td>Learn how to handle alerts on a CI that is in the maintenance state, which means it is undergoing some type of change.</td>
</tr>
<tr>
<td>Advanced task</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Adjust alert impact while triaging an alert</td>
<td>You can change impact rules, which affects how alert impact is calculated and alert severity.</td>
</tr>
</tbody>
</table>

**Customize your Event Management Dashboard view**

You can create one or more customized dashboard views that show only the information pertinent to you. For example, you might want to focus only on the alerts for business services in a specific location, or only the alerts for business services of high criticality.

1. Navigate to Event Management > Settings > Dashboard Views.
2. In the Dashboard Views screen, click New.
3. In the Name field, enter a descriptive name for the dashboard view.
4. Keep the Active option selected to use the dashboard view.
5. Specify the filter conditions, and then click Submit.

**Note:** The Dashboard View feature applies only to business services, and not to business service groups. The business service groups appear whether or not they match the filter, and whether or not they contain any services that match the filter.
In this example, the only alerts that can appear on your customized dashboard are those that belong to the most critical or somewhat critical business services in the U.S. datacenter.

6. Navigate to Event Management > Dashboard and verify that the view works as expected.

Handle alerts in the flapping state
If an alert is in the flapping state, you might need to triage the alert again.

If an event repeatedly triggers the same alert in a short amount of time, the alert is put into the flapping state, even if an operator closed it previously. The event can also be fluctuating between severity levels, such as Info and Critical. You should triage the alert again, and take action to try to prevent the event from reoccurring.
Note: Your Event Management administrator can configure several settings that determine when to mark an alert as flapping, and other settings that determine what action you can take. This topic covers a generic example.

1. Find alerts in the flapping state:
   a) Navigate to Event Management > Alert console.
   b) Modify the list of alerts by clicking the filter icon ( ) and specifying criteria to find flapping alerts. For example, you can specify: State is Flapping.
   c) Click Run to get the list of alerts in the flapping state.

2. Click the alert number to open it.
3. On the alert, click the Flapping tab and review the information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flap count</td>
<td>The number of times the alert started flapping since the time in the Flap</td>
</tr>
<tr>
<td></td>
<td>start window field.</td>
</tr>
<tr>
<td>Flap start window</td>
<td>The time that flapping started.</td>
</tr>
<tr>
<td>Flap last update time</td>
<td>The last time flapping occurred. This time is the ServiceNow instance</td>
</tr>
<tr>
<td></td>
<td>processing time, not the source system time.</td>
</tr>
<tr>
<td>Flap last state</td>
<td>The state of the alert before it entered the flapping state.</td>
</tr>
</tbody>
</table>

4. Decide what action to take based on how often flapping has occurred. You can do any of the following, depending on what your organization's standard operating procedures are and what your Event Management administrator configured:
   - Run a remediation workflow that might do something like reboot a server or reset a router.
   - Access a web application, like the application for your event monitoring software on your network.
   - Modify an associated incident or a KB article to include additional information or steps.

5. After you are certain that the underlying issue is addressed, close the alert.

Handle alerts while CIs are in maintenance
When an alert occurs on a CI that itself is in a maintenance state, the alert state is also changed to maintenance. You should find and monitor the states of these alerts and, when any changes on the CI are complete, finish triaging the alert and close it.

Before completing this task, familiarize yourself with basic form configuration in a ServiceNow instance. You might need to add fields and related lists to forms.

The example in this topic assumes a network administrator opened a change request to reconfigure a problematic web server.

1. Find alerts in maintenance:
   a) Navigate to Event Management > Alert Console.
b) Modify the list of alerts by clicking the filter icon ( ) and specifying criteria to find alerts in maintenance. For example, you can specify:

State | is | Maintenance.

c) Click Run to get the list of alerts in the maintenance state.

2. In the list of alerts, find an alert with a value in the Configuration Item column. This is the CI on which the alert occurred.

3. Click the CI name to open the form that displays information about the CI.

4. Configure the form layout and the related lists to add the Status field and the Change Request related list.

Verify that the Status field appears in the position you put it, and that Change Request related lists appears at the bottom.
In this example, the status of the web server is **In Maintenance** and there is a change request scheduled so that someone can reconfigure it.

5. At this point, you should not modify the alert associated with this CI. You can open the change request to see if a schedule is attached, or coordinate with your team to find out when the CI will no longer be in maintenance.

6. After you verify that the **Status** field on the CI is not **In Maintenance**, open the alert again and clear the **Maintenance** check box.

7. Continue to **triage the alert**.

### Adjust alert impact while triaging an alert

As an Event Management operator, you might need to modify the impact that an alert has on CIs in a business service and on the business service itself. Do this when you think that the impact on a business service does not accurately represent what you see in your network environment and so other operators and administrators see the correct impact information.

Before you make any modifications to impact calculations, make sure that you understand how the business services and the CIs involved are actually functioning in your organization. This might require you to perform tests on network infrastructure or to contact personnel who can do so.

1. Navigate to **Event Management > Dashboard**, and then double click the name of the business service that has the alerts you want to modify.

2. Click the **Impact Tree** button.

3. Analyze the business service to see if the impact of the alerts is accurately reflected in the Impact Tree based on what you know about the CIs involved.

   In this example, a critical alert indicates a serious issue with a Linux server that is part of a Web portal business service.
Notice that the impact on the Web portal itself is red, meaning critical, even though only one of the Linux servers that it relies upon actually has a critical alert. The other Linux Server does is not impacted by the alert. Assume for this example that the Linux servers perform different roles for the web service, and that even though PS LinuxApp02 has a critical alert, the PS LinuxApp01 server can adequately provide all the necessary services to keep the business service operational.

4. Click the PS Apache03 CI, and then click the Impact button under the business service map to display the impact rules for that CI.

By default, the CI impact rule appears, showing you this information:
• The **Impact On** and **Influence** columns show you that the CI (PS Apache03) should inherit 100% of the impact of all child CI impact, which in this case includes the critical impact on the **PS LinuxApp02** server. This setting is why the **PS Apache03** web server, and the business server as a whole, inherits the impact 100%.

• The **Critical** column shows you that a critical alert (red) is inherited as critical alert (red).

5. Change the **Influence** value to 0 since the web server is still operational, and then change the **Critical** column to indicate that a critical impact status (red) should be inherited as a non-impact (green).

6. Wait for the dashboard to update automatically, or click **Event Management > Dashboard** and double-click the same business service to get an updated business service map.

Notice that the influence on the Apache web server is 0% and the impact status is green. Also notice that the critical alert status (red) remains unchanged, even though the impact changed to green.
7. Navigate back to the Event Management Dashboard. Notice that the business service is now green, indicating that the overall business service is operational. The critical alert still displays at the bottom in the Alerts list, indicating that there is still an issue that you or other operators can address.
Your administrator might configure additional impact rules that display when you click on a CI in the business service map. You can make adjustments to the **Influence** fields on any impact rule to get the most accurate impact tree.

**Mobile experience for Event Management**

You can manage Event Management alerts and business services from anywhere using the Event Management mobile application.

The mobile interface displays alerts and services, each of which is organized by top and high severity and top and high priority. You can also view alert and service details, perform various actions on alerts and services, create incidents, and add work notes.

To work with the Event Management mobile app, you must configure the app on your mobile device. You can also view and configure mobile settings within the app.

The Event Management mobile app works on both Android and IOS devices. The app runs on the ServiceNow Agent app, which you install from the App Store. For details, see [ServiceNow mobile app configuration](#).
If you encounter incidents that are not available on the mobile app, ensure that you have installed the ITSM mobile plugin (com.sn_itsm_mobile). For information on plugins, see ServiceNow plugins.

Event Management mobile functionality is also available on the Event Management instance.

**Connect to an instance using the mobile app**

After installing the mobile app, you must configure an instance from which the app measures alerts. When connecting to an instance using the mobile app, you can access and manage Event Management alerts.

Role required: evt_mgmt_operator or evt_mgmt_admin

1. Configure the Event Management mobile app. For more information, see ServiceNow mobile app configuration.
2. In the ServiceNow AgentEvent Management mobile app, click the plus icon to add a new instance.
3. Select the instance you want to configure.
4. Log in with either user, operator, or admin credentials.

Monitor alerts or services in the mobile app, as described in Work with Event Management alerts using the mobile app and Work with Event Management services using the mobile app.

**Work with Event Management alerts using the mobile app**

You can perform a variety of actions on Event Management alerts in the Event Management mobile app.

*Acknowledge an alert using the mobile app*

Acknowledge an alert on your mobile device to indicate that the alert needs to be addressed.

Role required: evt_mgmt_operator or evt_mgmt_admin

1. Open the mobile app and select Event Management. The home page appears.
2. Tap one of the alert groups.
   The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
   - Priority
   - Time of update
3. **Slide an alert to the left and tap Ack on the right side of the alert.**
   The alert's Acknowledged state is set to True.

---

°2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Place an alert into maintenance using the mobile app
Place an alert into maintenance on your mobile device to indicate that the alert's CI is in the Maintenance state.

Role required: evt_mgmt_operator or evt_mgmt_admin

1. Open the mobile app and select Event Management. The home page appears.

2. Tap one of the alert groups. The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
- Priority
- Time of update

**Note:** The procedure for working with priority and severity alerts is the same.

3. Slide an alert to the right and tap **Maint** on the left side of the alert.
The alert is set to the Maintenance state and displays with both a blue indicator in the left margin and diagonal lines running through the alert. After refreshing the page, this alert no longer appears in the list of alerts.

**View repeated alerts using the mobile app**
You can check whether the issue tracked by your alert on your mobile device has occurred multiple times in the system to help determine how urgently the alert needs to be dealt with. Repeated alerts have the same source and the same severity as a particular alert.

**Role required:** evt_mgmt_user

1. Open the mobile app and select Event Management.
   The home page appears.

2. Tap one of the alert groups.
The relevant grouping of alerts appears, displaying the following alert details:

- Number
- Description
- Severity
- Priority
- Time of update
Note: The procedure for working with priority and severity alerts is the same.

3. Select an individual alert.
4. Tap the More Options icon (•••) and select **Repeated Alerts**. A list is displayed containing the alerts with the same source and severity as the selected alert.

*View similar alerts using the mobile app*
Check whether the CI tracked by your alert on your mobile device has occurred multiple times in the system to help determine how urgently the alert needs to be dealt with. Similar alerts are alerts with the same CI as a specified alert but do not necessarily match all aspects of the alert. Similar alerts are identified by Machine Learning (ML) discovering similar text in alert fields.

**Role required:** evt_mgmt_user

1. Open the mobile app and select Event Management. The home page appears.
2. Tap one of the alert groups.
   The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
   - Priority
   - Time of update
3. Select an individual alert.

Note: The procedure for working with priority and severity alerts is the same.
4. Tap the More Options icon ( ■ ) and select **Similar Alerts**. A list is displayed containing alerts similar to the selected alert.

*View incidents related to alert CIs using the mobile app*

Viewing a list of alert CIs for which an incident is open helps determine which CIs are problematic in your Event Management environment.

**Role required:** evt_mgmt_user

1. Open the mobile app and select Event Management. The home page appears.
2. Tap one of the alert groups.
   The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
   - Priority
   - Time of update
Note: The procedure for working with priority and severity alerts is the same.

3. Select an individual alert.
4. Tap the More Options icon (⋮) and select **CI Incidents**. A list of the CIs of the alert that have related open incidents appears.

*View change requests related to an alert CI using the mobile app*

When viewing an alert in a mobile environment, you can view any change requests that exist on the CI associated with the current alert. A change request indicates when a specific CI will be unavailable.

**Role required:** evt_mgmt_user

Change requests provide a method to implement a controlled process for the addition, modification, or removal of approved and supported configuration items (CIs). A change request records the detailed information about the change, such as the reason of the change, the priority, the risk, the type of change, and the change category.

1. Open the mobile app and select Event Management.
   The home page appears.
2. Tap one of the alert groups.
   The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
   - Priority
   - Time of update
Note: The procedure for working with priority and severity alerts is the same.

3. Select an individual alert.
4. Tap the More Options icon ( ) and select **CI Change requests**.
A list of the change requests that exist on the alert’s CI appears.

*View alert executions using the mobile app*
You can view alert executions for an alert in the mobile environment. Alert executions track the history of an alert by displaying actions performed on the alert, and the alert management rules executed on the alert either manually or automatically.

Role required: evt_mgmt_user

1. Open the mobile app and select Event Management.
The home page appears.
2. Tap one of the alert groups.
   The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
   - Priority
   - Time of update
3. Select the relevant group to display a list of alerts.
4. Select an alert.

**Note:** The procedure for working with priority and severity alerts is the same.
5. Tap the Related tab and select Alert Executions to view the alert management rules executed for the alert.

The number next to the Alert Execution link indicates the number of alert management rules executed on the alert.
View secondary alerts using the mobile app

When viewing an alert in a mobile environment, you can also view any secondary alerts in the specified alert's group. The alert containing a group of secondary alerts is called the primary alert.

Role required: evt_mgmt_user

1. Open the mobile app and select Event Management. The home page appears.

2. Tap one of the alert groups. The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
- Severity
- Priority
- Time of update

Note: The procedure for working with priority and severity alerts is the same.
3. Select an individual alert.
4. Tap the Related tab and select Secondary Alerts to view the secondary alerts for the primary alert's group.

The number next to the Secondary Alerts link indicates the number of secondary alerts that are grouped with the specified alert.
Manage alert actions using the mobile app
When viewing alerts on your mobile device, you can track actions performed on the alert and log additional actions by adding work notes.

Role required: evt_mgmt_user

1. Open the mobile app and select Event Management. The home page appears.

2. Tap one of the alert groups. The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
• Priority
• Time of update

Note: The procedure for working with priority and severity alerts is the same.

3. Select an alert to view its information.
The **Details** tab is selected by default and displays details on the alert.

**Alert0010140**

**JVM Memory usage exceeds 89%**

**Critical**

2019-05-13 17:50:08

**Description**

JVM Memory usage exceeds 89%

**State**

Open

**Source**

Nanios

[Close Alert]

[Quick Incident]
4. Tap the **Activity** tab to view work notes or other activities associated with the alert. You can also tap the green plus sign to add a new work note or perform an activity on the alert, such as adding an attachment.

5. Tap the green plus sign icon to add a new work note or perform an activity on the alert.
Your text appears in the list of Work notes.

View services impacted by alerts using the mobile app
Alerts impact business services, which can affect the way those services run in your environment. You can view the business services impacted by alerts that your mobile device monitors.

Role required: evt_mgmt_user

1. Open the mobile app and select Event Management. The home page appears.

2. Tap one of the alert groups. The relevant grouping of alerts appears, displaying the following alert details:
   - Number
- Description
- Severity
- Priority
- Time of update
Note: The procedure for working with priority and severity alerts is the same.

3. Select an individual alert.
4. Tap the More Options icon (⋮) and select **Impacted Services**.
   The services impacted by the alerts appear.
   For details on how alert impact is calculated, see [Alert impact calculation](#).

**Create an incident for a mobile alert**
When viewing an alert on your mobile device, you can create an incident for the alert.

Role required: evt_mgmt_operator or evt_mgmt_admin

1. Open the mobile app and select Event Management.
   The home page appears.
2. Tap one of the alert groups.  
The relevant grouping of alerts appears, displaying the following alert details:

- Number
- Description
- Severity
- Priority
- Time of update
Note: The procedure for working with priority and severity alerts is the same.

3. In the group’s list, select an alert to view its information.
The Details tab appears by default with details on the alert.

4. Tap Quick Incident to create an incident from the alert with the setting Severity = Major.
Close an alert using the mobile app

If an alert on your mobile device has already been handled or is no longer relevant, you can choose to close it.

Role required: evt_mgmt_operator or evt_mgmt_admin

1. Open the mobile app and select Event Management. The home page appears.

2. Tap one of the alert groups. The relevant grouping of alerts appears, displaying the following alert details:
   - Number
   - Description
   - Severity
- Priority
- Time of update

Note: The procedure for working with priority and severity alerts is the same.

3. Select an alert to view its information.
The Details tab appears by default with details on the alert.

4. Tap Close Alert to close the alert.
Work with Event Management services using the mobile app

You can view services and perform actions on services using the Event Management mobile app.

View service alerts and details using the mobile app

You can use the mobile app to view a service’s alerts, as well as its severity, priority, name, owner, and class.

Roles required: evt_mgmt_user, evt_mgmt_admin or evt_mgmt_operator

1. Open the mobile app and select Event Management.
2. Tap one of the Services groups.

The services in the selected group are listed with the following information:

- Name
- Owner
3. Tap a service to view further details.

4. Tap the Alerts button to view the alerts associated with the service. The service's current alerts are displayed.

**Note:** The mobile app displays only alerts from the em_alert table, which have not yet undergone impact calculation. Therefore, alert lists might be different on the mobile app than in the instance, where the list includes alerts from the em_history_alert table that have undergone impact calculation.
Receive push notifications for a service using the mobile app
You can choose to receive an update when a service’s status changes to Urgent or High. A notification will be displayed in the Event Management mobile app.

Role required: evt_mgmt_user

1. Open the mobile app and select Event Management.
2. Select the bell icon ( ) at the bottom of the home page.
   The Enable Notifications indicator appears.

3. Slide the notification indicator to the right to enable receiving notifications of services whose status changes to Urgent or High.

Operational Intelligence

ServiceNow® Operational Intelligence provides the ability to capture, and then explore and analyze operational metrics data, identifying and indicating anomalies. Operational Intelligence generates anomaly alerts that can be promoted to IT alerts and appear on the Alert Console and service health dashboard. You can leverage this analysis to prevent potential service outages.

Starting with the New York release, Operational Intelligence is part of ITOM Health in the IT Operations Management product.

Explore
- Upgrade to New York
- Understanding Operational Intelligence
- Domain separation and Operational Intelligence

Set up
- Get started
- MID Server and MID Server distributed cluster
- Configure metric connector instances
- Configure data source registration
- Choose metrics to monitor
- Create an event rule to map metrics to specific CI
- Use REST endpoint to push metric data from an external source

Administer
- Configure anomaly score thresholds
- Create configuration settings rule
- Disable a metric for a CI
- Specify custom upper and lower metric bounds
- Create anomaly alert promotion rule
- Create anomaly alert promotion rule blacklist
- Configure metrics for Agent Workspace

Use

Develop

Troubleshoot and get help

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Understanding Operational Intelligence

Use Operational Intelligence to identify and prevent potential service outages. Operational Intelligence, based on historical metric data, indicates anomalous behavior of CIs which events might not capture. Anomaly alerts can be promoted to regular IT alerts and appear on the Alert Console and service health dashboard for preventive actions.

Starting with the New York release, Operational Intelligence is part of ITOM Health in the IT Operations Management product.

Anomaly detection

Metric data is collected by various data sources such as SCOM, SolarWinds monitoring system, or Nagios XI server (some partially configured for metric collection by default). These monitoring systems collect metric data from the source environment regularly. Operational Intelligence captures the raw data from these monitoring systems, and uses event rules and the CMDB identification engine to map data to existing CIs and their resources. The data is then analyzed to detect anomalies and to provide other statistical scores.

Operational Intelligence uses historical metric data to build statistical models. These models facilitate projection of expected metric values along with upper and lower bounds. Operational Intelligence then uses these projections to detect statistical outliers and to calculate anomaly scores. Anomalies are scored on a range of 0-10. High anomaly scores for CI metrics can indicate that a CI is at risk of causing a service outage.

After processing, the Insights Explorer shows metric statistics and charts, and the Anomaly Map shows correlated scores for CIs with the highest anomaly scores, across a timeline.

Operational Intelligence is available when you activate the Operational Intelligence (com.snc.sa.metric) plugin.

Terms used with Operational Intelligence

Source metric type

A metric such as ‘% Free Space’ or ‘Current Bandwidth’ that can be measured by a data source for a CI. For each data source, you can choose which of all possible source metric types are processed. For example, there are about 380 source metric types that are active by default for the SCOM data source.

Anomaly
Data that is outside the control bounds is considered a statistical outlier. These outliers are used to compute an anomaly score, which is a value between 0–10 that indicates the degree to which the metric appears unlikely. When an anomaly score is above a threshold, an anomaly alert is generated. Anomaly alerts are reported separately from regular IT alerts.

**Resource**

A component of a CI that consists of multiple individual components of similar type, where each subcomponent can be monitored separately. For example, individual Web pages, or specific disks such as 'Disk C:' and 'Disk D:'.

**Time series**

A series of values (such as metric values) over a time range, associated with a CI and a metric type. Because an anomaly score is evaluated for each metric, the series of anomaly scores over a period of time are also a time series. Time series are computed by the statistical model built for a metric data series, and are used with metric data values, anomaly scores, and upper and lower control bounds.

**Statistical models**

Operational Intelligence jobs learn from past metric data (up to 32 days old). A model training process analyzes historical data to construct a model that projects future values. Typically, models are in effect until the next time the model learning process runs. These models are used to calculate upper and lower bounds. Incoming values that are beyond those bounds, and that deviate with statistical significance from expected values, generate anomalies. Each model is uniquely patterned and is labeled with a classifier that illustrates the general behavior of the model. This classification determines if anomaly detection can be applied. For most models, it is possible to project which future values deviate from expected values. Such models are associated with control bounds and anomaly detection can be applied (if enabled).

However, for some models, there is insufficient data to determine which values are anomalous and anomaly detection cannot be applied without additional information (even if anomaly detection is enabled).

The learned data models are stored in the Metric Time Series Models (sa_time_series) table.

The following statistical models and classifiers are used in anomaly detection:

**Time Series statistical model**

After it is established, a time series model does not adjust to changes in the incoming metric data. Therefore, if the pattern of incoming data changes, those changes are likely to be identified as anomalous. Upper and lower control bounds, after they are learned, persist until the next time the learning process runs (data is learned every day).

**Weekly**

Data with a pattern that repeats itself over weekly intervals (seasonal model).

Requires a minimum of 15 days of data in the series, as set by the `weekly_model_min_days` configuration setting.
Daily

Data with a pattern that repeats itself over a daily interval (seasonal model).

Requires a minimum of 3 days of data in the series, as set by the `daily_model_min_days` configuration setting.

---

Trendy

Data that has a linear trend with some slope and with some noise.

Requires a minimum of 30 data points in the series, as set by the `corrupt_data_count_threshold` configuration setting.

---

Noisy

Typical noisy data that is a basic pattern classification in a data model. The pattern cannot be identified with a specific trend or seasonality.
Requires a minimum of 30 data points in the series, as set by the `corrupt_data_count_threshold` configuration setting.

**Positive clipped noisy**

Similar to the noisy classifier other than the lower bound that is fixed on 0.

Requires a minimum of 30 data points in the series, as set by the `corrupt_data_count_threshold` configuration setting.

**Centered noisy**

Noisy data that typically spreads symmetrically between user-specified upper and lower bounds. The formula that is used to set bounds and width values, ignores the statistical data, and the lower and the upper widths have an identical value.

Requires that the number of data points in the series is zero.

See [Specify custom upper and lower metric bounds](#) for more information.
Skewed noisy

Noisy data that is not evenly spread between user-specified upper and lower bounds, but instead tends to concentrate closer to one of the bounds. The median of the data is used to separately compute an upper width and a lower width.

Requires a minimum of one data point in the series.

See Specify custom upper and lower metric bounds for more information.

Accumulator

Data pattern similar to the trendy classifier but with a monotonous increase and without noise. For this classifier, there is no data model and no anomaly detection.

Requires a minimum of 30 data points in the series, as set by the corrupt_data_count_threshold configuration setting.
Near Constant

Nearly constant data, in which most values are a specific constant value. For this classifier, there is no data model and no anomaly detection.

Requires a minimum of 30 data points in the series, as set by the `corrupt_data_count_threshold` configuration setting.

![Graph showing nearly constant data](image)

Multinomial

Data pattern in which all values are one of a relatively small number of values. For example, values are always 100 or 99.9. For this classifier, there is no data model and no anomaly detection.

Requires a minimum of 400 data points in the series, calculated as 10 times the value of the `multinomial_count_threshold` configuration setting.

![Graph showing multinomial data](image)

Corrupt

Data has insufficient data points to identify a pattern. For this classifier, there is no data model and no anomaly detection.

Requires that the number of data points in the series is less than the value of the `corrupt_data_count_threshold` configuration setting (30 by default).

![Graph showing corrupt data](image)

Kalman Filter statistical model

Add on to the time series statistical model and applicable only to the noisy and positive noisy classifiers. This model is a general method of estimating model parameters from a stream of data where level is the only parameter in the model. The Kalman Filter model can adjust to new values...
in incoming metric data. When there are no clear patterns in the noise or if there is too much noise, the Kalman Filter model is not used.

Local level
When incoming data clusters around a new value according to the current control bounds, the Learner adjusts the data model to accommodate a permanent change. This clustering is detected as a new value in the data model so that most incoming data is again within the control bounds rather than anomalous. Such change detection is useful when for example, cores or memory are added to a server, which impact the baselines.

Requires a minimum of 30 data points in the series, as set by the `corrupt_data_count_threshold` configuration setting.

Unrecognized
When data does not fit the local level classifier, time series classifiers are used. This happens when it is not possible to adjust the variance ratio in a learned local level model to reasonable values.

Non-Parametric statistical model
Add on to the positive noisy classifier. In the nonparametric model, noise distribution is not symmetrical and does not fit any seasonal pattern. The nonparametric model creates control bounds that better fit the actual data, and once learned, the control bounds persist until the next learning cycle. This model does not adjust itself to changes in the data, and it takes longer for a deviation to be identified as an anomaly.

Stationary Non-Parametric
Data that is not time-dependent meaning that there is no significant shift in parameters such as mean and variance when shifting data in time.

Requires a minimum of 5000 data points in the series, as set by the `snpm_minimum_data_count` configuration setting.
Unrecognized
When data does not fit the stationary classifier, time series classifiers are used.

Get started with Operational Intelligence

Complete the following setup and initial configuration steps to start using Operational Intelligence.

Raw metric data, collected by data sources (such as SCOM) needs to reach the Operational Intelligence MID Server, where Operational Intelligence can process and analyze it. You can set up Operational Intelligence to receive raw metric data from external data sources using either pull or push mechanism.

Using either method (pulling or pushing) requires the configuration of the Operational Intelligence Metrics MID Server extension. The main roles of this extension are to normalize the raw data, and to detect any anomalies. The normalized data and anomalies (if above the specified threshold), are then transmitted to the instance.

Some of the steps for setting up Operational Intelligence, must be completed regardless of the method used. Other steps depend on the method used for transmitting data between the data source and the MID Server:

Pulling
This method requires using a connector that is configured to pull data from a specific data source that is collecting raw metric data. You can use a pre-defined connector such as SCOM (which was previously defined for pulling events from SCOM), Azure platform, or AWS, or configure a new custom connector specifically for the external source that is used in your environment.

Pushing
This method uses web service APIs and is based on client-side tools that push the raw data from the external source to the MID Server. This method requires that the Operational Intelligence Metrics extension is configured with the Enable REST Listener option enabled.

You also need to configure the Web Server extension, which starts a Web Server on the MID Server. Request handlers are added on the Web Server that listen to any incoming metric data that was pushed from the external source.

Watch the Event Management | Collecting Events and Metrics short video about using MID Servers for pulling and pushing metric data.
In an initial setup, you can use an automated setup to automate many of the setup tasks. For example, to create a MID Server distributed cluster that contains a single Operational Intelligence MID Server. The following video describes the manual steps for setting up Operational Intelligence.

Operational Intelligence guided setup provides a sequence of tasks that help you configure Operational Intelligence on your ServiceNow instance. To open Operational Intelligence guided setup, navigate to Guided Setup > ITOM Guided Setup, and click Continue. Then scroll to Operational Intelligence, and click Continue. For more information about using the guided setup interface, see Using guided setup.

Alternatively, you can complete the following procedure.

1. **Activate the Operational Intelligence plugin.**
2. Prepare a MID Server to be configured for Operational Intelligence. For information, see MID Server and MID Server distributed cluster for Operational Intelligence.
3. Use the Automated Operational Intelligence setup.
4. To configure for pushing:
   a) Create the client scripts that push the raw data from the external source to the MID Server.
5. To configure for pulling, for each data source that you want to use for collecting metric data:
   a) Navigate to Event Management > Connector Definitions to determine if there is a pre-defined connector definition for the data source, and then create a new connector definition if need to (The pre-defined SCOM connector for example — is configured for pulling events from SCOM).
   b) Configure a metric connector instance for the data source (For a SCOM connector for example — see Configure the SCOM connector instance).
6. **Choose and configure metrics to monitor.**
7. Create event rules to map raw metric data to specific CI types and Create an event rule to bind metric events to host CIs.
8. About 10 minutes after the Operational Intelligence Metrics extension has started, run the quick start tests for Operational Intelligence to verify functionality.

Request Operational Intelligence

Operational Intelligence plugin (com.snc.sa.metric) requires a separate subscription and must be activated by ServiceNow personnel. This plugin includes demo data and activates related plugins if they are not already active.

See Get started with Operational Intelligence for the high level steps of setting up Operational Intelligence.

Role required: admin

To purchase a subscription, contact your ServiceNow account manager. The account manager can arrange to have the plugin activated on your organization’s production and sub-production instances, generally within a few days.

If you do not have an account manager, decide to delay activation after purchase, or want to evaluate the product on a sub-production instance without charge, follow these steps.

1. From your instance, navigate to System Definition > Plugins.
2. On the All Applications page, click Request Plugin to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>Date and time must be at least two business days from the current time.</td>
</tr>
<tr>
<td>Reason/Comments</td>
<td>Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments.</td>
</tr>
</tbody>
</table>

5. Click Submit.

**MID Server and MID Server distributed cluster for Operational Intelligence**

Using Operational Intelligence requires at least one MID Server distributed cluster which contains at least one MID Server that is configured for Operational Intelligence.

Even if using a single Operational Intelligence MID Server, that server must be added to a MID Server distributed cluster, and any additional MID Servers in that cluster, must be configured for Operational Intelligence. All Operational Intelligence MID Servers must be members of a MID Server distributed cluster. The first MID Server that is added to a distributed cluster, is designated as the Cluster Leader.

To support the specified throughput, create a distributed cluster with a single MID Server that meets the Operational Intelligence MID Server requirements. To support high availability, create a distributed cluster with two such MID Servers.

See [Get started with Operational Intelligence](#) for the high level steps of setting up Operational Intelligence.

**MID Server distributed clusters**

Activating Operational Intelligence (com.snc.sa.metric), automatically activates MID Server Distributed Cluster (com.snc.agent.distributed.cluster). MID Server Distributed Cluster supports the Distributed MID Server cluster type which is used exclusively with Operational Intelligence.
In a distributed cluster type, the MID Servers in the cluster communicate with each other to distribute and balance the work load. This communication requires that all MID Servers in a distributed cluster are able to communicate with each other. Ensure that the specified ports are open and that there are no firewalls that block communication.

**Operational Intelligence MID Server**

The Operational Intelligence MID Server supports the Operational Intelligence feature by processing data and detecting anomalies. The MID Server transmits batches of processed data to the instance. If an anomaly is detected, then information about the anomaly along with the raw data is sent immediately to the instance, regardless of the regular 10-minute cycle.

Each Operational Intelligence MID Server must be configured with the following settings:

- OperationalIntelligence as a supported application
- the Metrics capability
- running the Operational Intelligence Metrics extension

The OperationalIntelligence application that the Operational Intelligence MID Server is configured with, lets you add additional supported applications to the same MID Server. By default, Operational Intelligence is included in the definition of the MID Server ALL option. For information about modifying the behavior of the ALL option when selecting supported applications, see [Configure applications included in ALL Applications](#).

If Domain Support - Domain Extensions Installer is activated, then you can configure a MID Server with the Metrics capability, per domain. In this case, metrics for a business service are processed on the MID Server that is in the same domain as the business service. Otherwise, a MID Server from the global domain is used.

Ensure that the MID Server that you want to configure for Operational Intelligence:

- Is validated. For more information, see [Validate a MID Server](#)
- Is in running state
- Meets all software, hardware, and configuration requirements

### Hardware requirements for the virtual machine/host on which the MID Server runs (scaled for 5000 CIs with 100 metrics per CI)

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Supports approximately up to a total of 1.2 million metrics (12,000 CIs with 100 metrics per CI).</th>
<th>Supports approximately up to a total of 1.7 million metrics (17,000 CIs with 100 metrics per CI).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Three MID Servers, each has:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Memory: 12 GB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Processor: 4 cores</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Three MID Servers, each with:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Memory: 24 GB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Processor: 8 cores</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Software requirements

<table>
<thead>
<tr>
<th>Software</th>
<th>Supported versions</th>
<th>Additional requirements</th>
</tr>
</thead>
</table>
| Windows  | 32-bit and 64-bit versions:  
- Windows 2008 R2  
- Windows Server 2012 R2  
- Windows Server 2016 | 32-bit or 64-bit version of the MID Server |
| Linux    |  
- Red Hat Enterprise Edition Linux 6.6 or later  
- CentOS Linux 6.6 or later | 32-bit or 64-bit version of the MID Server |

Configuration requirements

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Requirement</th>
</tr>
</thead>
</table>
| MID Server service logon user (required only for pulling) | Must be set to a user with read access to the SCOM database (OperationsManagerDW).  
1. On the MID Server, open the MID Server Properties dialog box.  
2. Click the Log On tab.  
3. Select This account, and enter credentials for a user with the required access.  
4. Click Apply. |

Configuration Override

Update the wrapper-override.conf file in folder ../agent/conf:

- Add the following code at the bottom of the file:

  ```
  # GC
  wrapper.java.additional.5=-XX:+AlwaysPreTouch
  wrapper.java.additional.6=-XX:+UseG1GC
  wrapper.java.additional.7=-XX:+ScavengeBeforeFullGC
  wrapper.java.additional.8=-XX:+DisableExplicitGC
  # IP4
  wrapper.java.additional.9=-DpreferIPv4Stack=true
  ```

- Optionally, if a server has additional RAM available, for example 16 GB RAM, you can replace 4096 with 8192 in: wrapper.java.maxmemory=4096.

Automated Operational Intelligence setup

When you initially set up Operational Intelligence, you can automate many of the setup steps. For example, the automated setup configures a MID Server for Operational Intelligence and creates a MID Server distributed cluster which contains that MID Server.

Role required: evt_mgmt_admin
When the following conditions are met, the related link **Setup Operational Intelligence** appears on the MID Server form, enabling automated setup:

- The MID Server has been validated.
- The MID Server is in a running state.
- The property `sa.metric.display.mid.setup.action` is set to true (default value).

The Operational Intelligence automated setup performs the following setup tasks:

1. Configures the MID Server with the **OperationalIntelligence** application
2. Adds the **Metrics** capability to the MID Server
3. Creates a new MID Server cluster of type ‘distributed’ set with a default port of 47100 for Apache Ignite
4. Adds the MID Server to the new MID Server distributed cluster
5. Creates an Operational Intelligence Metrics Extension and starts it
6. Opens the Operational Intelligence Metrics Context form. On this form, you can click the **Setup REST Listener** UI action to create the MID Web Server extension configured with the ‘Enable REST Listener’ option and default port 8097

Use the **Setup Operational Intelligence** link only to create the first MID Server distributed cluster that contains a single MID Server, with default values. After the automated setup completes, you have to manually create any additional MID Server distributed clusters or add Operational Intelligence MID Servers.

1. Navigate to **MID Server > Servers**.
2. Click the MID Server that you want to configure as an Operational Intelligence MID Server.
3. On the MID Server form, click **Setup Operational Intelligence** in the Related Links section.
4. In the newly created Operational Intelligence Metrics Extension form, wait for **Status** to be ‘Started’. Then, if configuring the MID Server for pushing click the UI action **Setup REST Listener**. A notification appears about the MID Web Server extension being created. The **MID Web Server Extension** field appears and **Enable REST Listener** is checked. You can preview the MID Web Server Extension record to verify that its **Status** is ‘Started’.

**Manually configure a MID Server for Operational Intelligence**

To use Operational Intelligence, configure at least one MID Server with Operational Intelligence as a supported application, with the Metrics capability, and which runs the Operational Intelligence Metrics extension. Then, add that Operational Intelligence MID Server as a member to a MID Server distributed cluster.

Ensure that the MID Server meets all software, hardware, and configuration requirements to be configured for Operational Intelligence.

Role required: To access the MID Server - mid_server. To configure a MID Server in an instance (for example, to add a supported application), refer to the MID Server documentation.

1. Navigate to **MID Server > Servers**.
2. Click the MID Server that you want to configure for Operational Intelligence.
3. Add the **OperationalIntelligence** application:
   a) At the center of the MID Server form, click **Supported Applications**.
   b) In the **Supported Applications** section, click **Edit**.
   c) In the slushbucket, select **OperationalIntelligence** and click the > add button.
d) Click **Save**.

4. Add the Metrics capability:
   a) At the center of the MID Server form, click **Capabilities**.
   b) In the **Capabilities** section, click **Edit**.
   c) In the slushbucket, select **Metrics** and click the ‘>’ add button.
   d) Click **Save**.

5. Click **Update**.

Create a MID Server distributed cluster, and add the Operational Intelligence MID Server as a member to that cluster.

**Manually create a MID Server distributed cluster**

After creating at least one Operational Intelligence MID Server, create a MID Server distributed cluster for Operational Intelligence to contain that MID Server.

Role required: evt_mgmt_admin or agent_admin can manage cluster records

**Note:** If you used the related link Setup Operational Intelligence on a MID Server form, then you might not need to perform this procedure. If the automated setup completed successfully, then a MID Server distributed cluster which contains an Operational Intelligence MID Server, already exists. Use this procedure to manually create a MID Server distributed cluster and to add to it an Operational Intelligence MID Server. See ‘Automated setup of MID Server distributed cluster’ in this topic for more details.

Operational Intelligence uses two ports on each MID Server distributed cluster:

- One port for Discovery
- One port for communications

When deploying multiple Operational Intelligence MID Servers on the same physical server, allocate two ports for each additional MID Server. If for example, you deploy two Operational Intelligence MID Servers on the same physical server, then four ports are required.

1. Navigate to **MID Server > Clusters**.
2. Click **New**.
3. Fill out the MID Server Cluster form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the cluster.</td>
</tr>
<tr>
<td>Type</td>
<td>Cluster type which must be set to <strong>Distributed</strong>.</td>
</tr>
</tbody>
</table>

4. Right-click the form banner and select **Save**.
5. Additional fields appear on the MID Server Cluster form. Review the details, and then click **Update** if you modified any fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distributed Cluster</td>
<td>Indicates that the cluster type is distributed.</td>
</tr>
<tr>
<td>Status</td>
<td>The cumulative status of the cluster, which can be Started, Stopped, Error, or Warning.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Error Message</td>
<td>The most recent error message that was generated for the cluster.</td>
</tr>
<tr>
<td>Port</td>
<td>Starting port in the port range to use for communication with other Operational intelligence MID Servers in the distributed cluster. Default value is 47100.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If the same virtual machine/host server runs multiple Operational Intelligence MID Servers, and if those MID Servers are communicating with different ServiceNow instances or are part of a different distributed cluster on the same ServiceNow instance, then use separate ports for each cluster.</td>
</tr>
<tr>
<td>Port count</td>
<td>The number of ports to use for communication with other Operational Intelligence MID Servers in the MID Server distributed cluster. Port count is used with Port to determine a range of ports. If the value of Port is 47100, and the value of Port count is four, the ports 47100 - 47103 are used for communication. Default value is four. Do not modify this value if only a single MID Server is configured on a physical server. Otherwise, increase this value by two for each additional MID Server that is implemented on a single physical server.</td>
</tr>
</tbody>
</table>

6. In the Includes MID Servers related list, click Edit to add MID Servers to the cluster.
7. Use the slushbucket to add only MID Servers that were configured for Operational Intelligence. Then click Save.
8. On the MID Server Cluster form, click Update.

**Note:** The Operational Intelligence MID Servers do not join the MID Server distributed cluster until you configure the Operational Intelligence extension, which is the next step in setting up Operational Intelligence.

Parameters of MID Server distributed cluster
The following MID Server parameters support the MID Server Distributed cluster type.

You can access these parameters in the Configuration Parameters related list of a MID Server. To add and set any of these parameter, see Add a MID Server parameter.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid.distributed.cluster.striped.threadpool.size</td>
<td>Size of striped threadpool in MID Server distributed cluster.</td>
</tr>
<tr>
<td></td>
<td>• Type: integer</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: Number of cores on the virtual machine or server.</td>
</tr>
<tr>
<td>mid.distributed.cluster.log.level</td>
<td>Logging level for MID Server distributed cluster, which can help troubleshooting. However, logging impacts the performance of Operational Intelligence.</td>
</tr>
<tr>
<td></td>
<td>An empty value designates no logging.</td>
</tr>
<tr>
<td></td>
<td>• Type: integer</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: none</td>
</tr>
<tr>
<td></td>
<td>• Other possible values: info, debug, trace</td>
</tr>
<tr>
<td>mid.distributed.cluster.global.timeout</td>
<td>Global timeout in seconds for MID Server distributed cluster.</td>
</tr>
<tr>
<td></td>
<td>• Type: integer</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: 180</td>
</tr>
<tr>
<td></td>
<td>• Other possible values:</td>
</tr>
<tr>
<td></td>
<td>• Minimum value: 60 (not enforced)</td>
</tr>
<tr>
<td></td>
<td>• Maximum value: 300 (not enforced)</td>
</tr>
<tr>
<td>mid.distributed.cluster.node.max.off.heap.memory</td>
<td>The amount of off heap memory used by Operational Intelligence to process data series per minute.</td>
</tr>
<tr>
<td></td>
<td>The default 4 GB of space is the amount of off heap memory needed for processing 800,000 data series per minute. Adjust this value proportionally to the amount of actual data series. For example, to process 400,000 data series per minute, you can change the value to 2048.</td>
</tr>
<tr>
<td></td>
<td>• Type: integer</td>
</tr>
<tr>
<td></td>
<td>• <strong>Default value</strong>: 4096</td>
</tr>
</tbody>
</table>

**Scheduled jobs included with MID Server distributed cluster**

The following MID Server scheduled job is included with the MID Server Distributed cluster type.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validate MID distributed cluster</td>
<td>Validates status of the cluster node for each MID Server in the cluster. For a MID Server that is down and its status is not stopped, updates the node status to stopped.</td>
</tr>
<tr>
<td></td>
<td>Also, validates overall status of the cluster. During this validation, the job stops the cluster if there are multiple sub-clusters in a single cluster.</td>
</tr>
</tbody>
</table>
Manually configure the Operational Intelligence extension

Configure the MID Server Operational Intelligence extension to enable the MID Server to pull raw metrics from external systems, to detect anomalies and report anomalies to the instance along with raw data. This MID Server Operational Intelligence extension is required and must be running in order for your system to be able to collect Operational Intelligence data.

See [Get started with Operational Intelligence](#) for the high level steps of setting up Operational Intelligence.

Role required: evt_mgmt_admin

The MID Server Operational Intelligence extension normalizes the raw data and then transmits the data to the instance. If the extension detected any anomalies above a specified threshold, they are sent to the instance. The extension runs for as long as it is enabled. This provides a persistent connection to the MID Server to constantly listen for raw Operational Intelligence data from external systems.

The MID Server Operational Intelligence extension:

- Receives raw metric data, batches them and sends them to the instance at specified intervals.
- Detects anomalies and sends a report to the instance.
- The information that is sent to the ServiceNow instance is relevant for Operational Intelligence, so data processing is efficient.

The MID Server Operational Intelligence extension does not provide any API calls. However, when the **Enable REST Listener** option is selected, the extension adds a handler for the supported REST APIs.

**Note:**

- Only upon configuring the Operational Intelligence extension, the Operational Intelligence MID Servers actually join the distributed cluster for Operational Intelligence.
- After the initial configuration, the first metric is not included in the metrics data.
- There is a delay of one minute in receiving metric information from the synchronization of the instance with the MID Server.

1. Navigate to **MID Server > Extensions > Operational Intelligence**.
2. In the Operational Intelligence Metrics Contexts list, click **New**.
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for this extension for easy identification.</td>
</tr>
<tr>
<td>Short description</td>
<td>A description of this extension.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Status         | This field is auto-populated with the status of the extension. The field is blank until the extension is started. After issuing a command to the extension, one of the following values is displayed:  
- **Started**: The extension is running.
- **Stopped**: The extension is not running.
- **Offline**: The MID Server is down.
- **Error**: The extension failed with an error (the error message is displayed in Error Message).  
- **Warning**: A run-time exception has occurred. The extension continues to work. |
| Error Message  | Message describing any error that causes a command, such as Start or Stop, to fail. This field appears when the value in the Status field is Error and also when the value in the Status field is Warning. |
| Execute on     | Location for running this extension. When configuring this option for use with Operational Intelligence, set to Specific MID Server. Then specify the MID Server with Operational Intelligence as a supported application and the Metrics capability. |
| MID Server     | The name of the designated MID Server, which is configured with Operational Intelligence as a supported application and the Metrics capability. |
| Executing on   | (Read-Only) The name of the MID Server on which the extension is running. This field shows the name of the MID Server even if the MID Server is down. If the user stops the extension, this field is empty. |

4. When using the Push method for collecting Operational Intelligence data, the MID Server Operational Intelligence extension must be configured with the Enable REST Listener enabled. This option enables a listener so that a REST endpoint can receive raw metric data. The raw metric data is then placed in the regular data flow where the data is sent to the instance and the anomaly detector looks for anomalies. When selected, it adds a handler to the web server to listen for any metrics that are pushed to the MID Server. When this option is selected, the Web Server extension, which starts a Web Server on the MID Server, must also be configured. For more information, see Configure the MID Web Server extension.

5. Right-click the form heading and select Save.

6. Under Related Links click Start to save the Operational Intelligence data in this extension and start the extension.

#### Commands available in the MID Server Operational Intelligence extension

<table>
<thead>
<tr>
<th>Related Link</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>Starts the extension on the configured MID Server if it is currently not running.</td>
</tr>
</tbody>
</table>
### Manually configure MID Web Server for metric data

Configure the MID Web Server extension to enable external clients to push metric data to the MID Server. This extension is used to listen for raw metric data as well as external events and it provides options for authentication and data security.

- Deploy and start a MID Server.
- The **Enable REST Listener** option must be selected in the Operational Intelligence Metrics extension. For more information, see [Manually configure the Operational Intelligence extension](#).
- If the **Secure Connection** option is going to be selected, first obtain a server certificate. For more information, see [Setup certificate for secure connection](#).

Role required: evt_mgmt_admin

You can configure the MID Web Server to collect raw metric data by the MID Server and then transmit it to the instance. The MID Web Server extension runs for as long as it is enabled. The extension starts a web server on the MID Server to serve web requests from external systems. The raw data is pushed to the extension from a client or by using customized script.

1. Configure a MID Web Server. For more information, see [Configure the MID Web Server extension](#).
2. Configure the authentication and secure connection settings, as required.
   - **Authentication type** can be set to the more advanced option — **Keybased**.
   - **Secure Connection** lets you choose whether incoming and outgoing data is secured when transmitted. If you choose the more advanced secured option, it requires that you obtain a certificate from a well-known certificate authority, and then provide the **Keystore Certificate Alias** and the **Keystore Password**.
3. Click **Save** to save the Operational Intelligence data.
4. Under **Related Links** click **Start** to start the collector.

#### Commands in the MID Web Server form

<table>
<thead>
<tr>
<th>Related Link</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop</td>
<td>Stops the running extension on the configured MID Server. No action is taken if the extension is not running.</td>
</tr>
<tr>
<td>Restart</td>
<td>Stops, then starts the extension on the configured MID Server.</td>
</tr>
<tr>
<td>Test</td>
<td>The test is not relevant to Operational Intelligence. Parameters are not tested or validated when <strong>Test</strong> is run.</td>
</tr>
<tr>
<td>Update parameters</td>
<td>Sends the latest saved parameters to the extension.</td>
</tr>
<tr>
<td>Related Link</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Test parameters</td>
<td>The test is not relevant to Operational Intelligence. Parameters are not tested or validated when Test is run.</td>
</tr>
<tr>
<td>Update parameters</td>
<td>Stops and then starts the MID Web Server with new parameters. If none of the parameters are modified, no update is made.</td>
</tr>
</tbody>
</table>

Create an authentication token to be sent with each request (Keybased authentication type).

**Method:** Create a token by constructing a string using defined elements of the HTTP/HTTPS request. Then create an HMAC (Hash Message Authentication Code) of the string by signing the generated string with the auto-generated secret key that is displayed in the Secret Key. This key is unique per context. Send this authentication token in the request header Authorization.

Data for the example:

**Keybased authentication data**

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path to a web service API for sending raw data</td>
<td>URL format: https://&lt;MID Server IP address&gt;:&lt;port number&gt;/api/mid/sa/metrics Use a port number that matches one of the port numbers setup in the Web Server extension. Example: <a href="http://10.10.10.10:8097/api/mid/sa/metrics">http://10.10.10.10:8097/api/mid/sa/metrics</a></td>
</tr>
<tr>
<td>Request type</td>
<td>POST</td>
</tr>
<tr>
<td>Date format</td>
<td>yyyy-MM-dd'T'HH:mm:ss.SSS'Z'</td>
</tr>
<tr>
<td>For example:</td>
<td>2016-06-08T20:54:58.917Z</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
</tbody>
</table>

Use the following request elements to generate the required string: HTTP-Verb, Content-Type, Date, and request path. Specify these elements and place them in this order:

- HTTP-Verb + "\n" +
- Content-Type + "\n" +
- Date + "\n" +
- Request-Path

For this example, the request string is:
For the timestamp requirement, a valid timestamp that uses HTTP date header is required for authenticating the request. Ensure that the timestamp is within 15 minutes of the MID Server.

**Example, using Java, that describes how to generate the HMAC of the string that uses defined elements of the HTTP/HTTPS request.**

```java
package sample;
import com.glide.util;
import java.security.SignatureException;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;

public class AuthUtil {

private static final String HMAC_SHA1_ALGORITHM = "HmacSHA1";

    /**
     * Generates base64-encode the HMAC(Hash Message Authentication Code) of input data
     *
     * @param data
     * @param key
     * @return
     * @throws java.security.SignatureException
     */
    public static String signData(String data, String key) throws java.security.SignatureException {
        String result;
        try {
            // get an hmac_sha1 key from the raw key bytes
            SecretKeySpec signingKey = new SecretKeySpec(key.getBytes(), HMAC_SHA1_ALGORITHM);

            // create hmac_sha1 Mac instance and initialize with the signing key
            Mac = Mac.getInstance(HMAC_SHA1_ALGORITHM);
            mac.init(signingKey);

            // compute the hmac on input data bytes
            byte[] rawHmac = mac.doFinal(data.getBytes("UTF-8"));

            // base64-encode the hmac
            result = Base64.encode(rawHmac);
        } catch (Exception e) {
            throw new SignatureException("Failed to generate HMAC : " + e.getMessage());
        }
        return result;
    }
}
```

**Setup certificate for secure connection**

To use a secure connection, obtain and configure a server certificate.
You must import the server certificate before you can add it to the Java KeyStore.

1. Add a server certificate to the Java KeyStore.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you have the RSA private key in the Java KeyStore and generated the certificate from that key.</td>
<td>keytool -import -alias keyname -file server.cert -storetype JCEKS -keystore webserver_keystore.jceks -storepass pwd</td>
</tr>
</tbody>
</table>

If you have a PKCS12 file that contains the RSA key and the certificate.

| Description | keytool -importkeystore -destkeystore webserver_keystore.jceks -deststoretype jceks -srckeystore <PKCS12 filename> -srckeystore pkcs12 |

Ensure that the private key password is the same as the Java Keystore password.

To change the password, use this command:

    keytool -keypasswd -keystore webserver_keystore.jceks -alias <key alias>

For testing, you can use this command to generate a self-signed certificate.

    keytool -genkeypair -alias webserver -dname cn=localhost -validity 365 -keyalg RSA -keysize 1024 -keypass mykeypass -storetype jceks -keystore webserver_keystore.jceks -storepass mykeypass

2. Save the webserver_keystore.jceks file under the /keystore directory.

The user must provide the certificate alias and password when configuring the MID Web Server extension. For more information, see Configure the MID Web Server extension.

**Quick start tests for Operational Intelligence**

Validate that Operational Intelligence still works after you make any configuration change such as apply an upgrade or develop an application. Copy and configure these quick start tests to pass when using your instance-specific data.

Operational Intelligence quick start tests require activating the Operational Intelligence plugin (com.snc.sa.metric).
<table>
<thead>
<tr>
<th>Test</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OI: Health Metrics Collection</td>
<td>An individual test that verifies the following:</td>
</tr>
<tr>
<td></td>
<td>· There is at least one Operational Intelligence Metrics extension that is running.</td>
</tr>
<tr>
<td></td>
<td>· There are entries for the ‘Health Metrics’ in the Metric to CI table.</td>
</tr>
<tr>
<td></td>
<td>· For each of the Operational Intelligence Metrics extensions that are currently running, that there is data stored in MetricBase.</td>
</tr>
<tr>
<td></td>
<td>If the test initially fails, wait until the Operational Intelligence Metrics extension runs for at least 10-15 minutes. Then try again.</td>
</tr>
</tbody>
</table>

**Components installed with Operational Intelligence**

Several types of components are installed with activation of the Operational Intelligence (com.snc.sa.metric) plugin, including tables, scheduled jobs, and properties.

*Note:* To view all other components that install with this application, see the Application Files table. For instructions on how to access this table, see Find components installed with an application.

**Scheduled jobs installed**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Intelligence — Metric Learner job</td>
<td>Runs daily and builds the statistical models used for anomaly detection.</td>
</tr>
<tr>
<td>Operational Intelligence — Purge old metric schemas and anomaly scores</td>
<td>Cleans up the sa_metric_anomaly_score and the sa_metric_schema_definition tables.</td>
</tr>
<tr>
<td>Operational Intelligence — Sync metric schema to mid</td>
<td>Synchronizing data from the instance to the MID Server in the metric schema table sa_metric_schema_definition.</td>
</tr>
<tr>
<td>Operational Intelligence — Sync tables with mid</td>
<td>Synchronizing data from the instance to the MID Server in tables.</td>
</tr>
<tr>
<td>Operational Intelligence — Table cleanup</td>
<td>Cleaning up tables related to various deletion operations.</td>
</tr>
<tr>
<td>Operational Intelligence — Observation Interval Learner job</td>
<td>Calculates the observation interval using collected data points for all data series.</td>
</tr>
<tr>
<td>Operational Intelligence — Process Stale Anomaly Score – Daily</td>
<td>Daily job that cleans stale anomaly scores, for a specified window of time, in the Metric Anomaly Score (sa_metric_anomaly_score) table. Uses the properties sa_metric.anomaly_score_stale_interval_in_days and sa_metric.anomaly_score.next_stale_interval_in_days to calculate the window of time for cleaning.</td>
</tr>
<tr>
<td>Operational Intelligence — Sync CI Anomaly Scores</td>
<td>Keeps anomaly scores up to date with any deletion of metrics.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Operational Intelligence — Metric configuration job</td>
<td>Synchronizes metric configurations from the instance to all Operational Intelligence MID Servers.</td>
</tr>
<tr>
<td>Operational Intelligence — Local level kalman filter model learner job</td>
<td>Builds local level kalman filter model learner job which runs when user clicks Run Anomaly Test.</td>
</tr>
<tr>
<td>Operational Intelligence — Anomaly Model Test</td>
<td>On-demand scheduled job which runs when user clicks Run Anomaly Test.</td>
</tr>
<tr>
<td>Operational Intelligence — Update Metric CI Type Map</td>
<td>Goes through the different data sources that are used in Advanced Insights Explorer and synchronizes the CI Type to Metric Type Mapping for each of these sources. Runs daily.</td>
</tr>
</tbody>
</table>
| Operational Intelligence — Validate MID Distributed cluster | 1. Checks the status of the cluster node. If status is Started and the status of the Operational Intelligence extension is not any of Started/Starting/Warning (warning with specific errors), then the job changes the status of the cluster node to Stopped. This happens when MID Server stops unexpectedly without notifying the instance that it is stopping.  
2. Checks the status of all cluster nodes on all MID Servers in the cluster.  
   - If the status of all nodes is Stopped and the status of the cluster itself is Started, then the job changes the status of the cluster to Stopped.  
   - If the status of the cluster is Stopped but there are MID Servers with a cluster node that is running, then the job changes the status of the cluster to Started.  
3. Checks the number of Cluster Leaders: If two are detected, then the job stops the entire cluster. This happens when MID Servers are not able to connect with other members of the cluster and form multiple sub-clusters. |
<p>| Event Management - process metric binding events (1000-1020) | Process Operational Intelligence metric binding events separately from Event Management events (bucket range 1000-1020). |
| Event Management - process metric binding events (1020-1040) | Process Operational Intelligence metric binding events separately from Event Management events (bucket range 1020-1040). |
| Event Management - process metric binding events (1040-1060) | Process Operational Intelligence metric binding events separately from Event Management events (bucket range 1040-1060). |
| Event Management - process metric binding events (1060-1080) | Process Operational Intelligence metric binding events separately from Event Management events (bucket range 1060-1080). |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Management - process metric binding events (1080-1100)</td>
<td>Process Operational Intelligence metric binding events separately from Event Management events (bucket range 1080-1100).</td>
</tr>
<tr>
<td>Operational Intelligence - Stationary nonparametric model learner job</td>
<td>Learns the stationary nonparametric model.</td>
</tr>
<tr>
<td>Operational Intelligence - Combined Model Learner Job</td>
<td>Constructs all the information required for performing real time anomaly detection (calculating data frequency and building statistical models) on a subset of all series for which data is being collected.</td>
</tr>
</tbody>
</table>

**Tables installed**

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric Time Series Model (sa_time_series)</td>
<td>Statistical models built for metric data.</td>
</tr>
<tr>
<td>Alert Anomaly (em_alert_anomaly)</td>
<td>Anomaly alerts that were created from anomaly events that were sent by Operational Intelligence so users can review. Anomaly alerts are kept separately from regular system alerts.</td>
</tr>
<tr>
<td>Metric Anomaly Score (sa_metric_anomaly_score)</td>
<td>Stores anomaly scores for the metric.</td>
</tr>
<tr>
<td>Metric To CI Mapping Deleted SysIds (sa_metric_map_deleted)</td>
<td>SysIds of deleted metric-to-CI mapping records.</td>
</tr>
<tr>
<td>Metric To CI Mapping (sa_metric_map)</td>
<td>Mappings currently in effect of metric types to CIs.</td>
</tr>
<tr>
<td>SA Metric Type Registration (sa_metric_registration)</td>
<td>Details about metric type registration.</td>
</tr>
<tr>
<td>Metric Type Registration Deleted SysIds (sa_metric_registration_deleted)</td>
<td>SysIds of records that were deleted from the SA Metric Type Registration (sa_metric_registration) table.</td>
</tr>
<tr>
<td>Metric Schema Definition (sa_metric_schema_definition)</td>
<td>Map of metrics being received based on CI class. It is used to optimize the metric data payload being sent from the MID Server to the instance.</td>
</tr>
<tr>
<td>Metric Type (sa_metric_type)</td>
<td>Metric type source.</td>
</tr>
<tr>
<td>Monitoring System Metric Type Deleted SysIds (sa_source_metric_type_deleted)</td>
<td>SysIds of deleted monitoring system metrics.</td>
</tr>
<tr>
<td>Monitoring System Metric Type (sa_source_metric_type)</td>
<td>Metric types per CI class, active/inactive status, and metric source.</td>
</tr>
<tr>
<td>Metric Bookmark (sa_metric_bookmark)</td>
<td>Saved user views/bookmarks for the Operational Intelligence Insights Explorer.</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Metric Settings</td>
<td>Default configuration settings for Operational Intelligence MID Servers.</td>
</tr>
<tr>
<td>(sa_metric_config_setting)</td>
<td></td>
</tr>
<tr>
<td>Metric Configuration Rules</td>
<td>Rules to override configuration settings for MID Servers.</td>
</tr>
<tr>
<td>(sa_metric_config_rule)</td>
<td></td>
</tr>
<tr>
<td>Metric overridden configurations</td>
<td>Maps overridden configurations to custom defined rules.</td>
</tr>
<tr>
<td>(sa_metric_config_rule_setting)</td>
<td></td>
</tr>
<tr>
<td>Metric Staged Configurations</td>
<td>Saved encoded configurations for the MID Servers.</td>
</tr>
<tr>
<td>(sa_metric_config_staging)</td>
<td></td>
</tr>
<tr>
<td>Metric learned parameters</td>
<td>Parameters learned for CI/metric combinations, that will be synced to the MID Servers.</td>
</tr>
<tr>
<td>(sa_metric_learned_param)</td>
<td></td>
</tr>
<tr>
<td>Metric To CI Mapping Deleted Details</td>
<td>Details about deleted entries from metric to CI mapping table. Includes CI, metric, and SysId of the deleted entry in the Metric To CI Mappings (sa_metric_map) table.</td>
</tr>
<tr>
<td>(sa_metric_map_deleted_detail)</td>
<td></td>
</tr>
<tr>
<td>Track unique CIs created in metric map</td>
<td>All unique CIs for which metric map entry was created.</td>
</tr>
<tr>
<td>(sa_metric_ci_tracker)</td>
<td></td>
</tr>
<tr>
<td>Metric Class</td>
<td>Metric classes that can be associated with specific metrics to override bounds and widths learned values.</td>
</tr>
<tr>
<td>(sa_metric_class)</td>
<td></td>
</tr>
<tr>
<td>Metric Change Score</td>
<td>Change events information.</td>
</tr>
<tr>
<td>(sa_metric_change_score)</td>
<td></td>
</tr>
<tr>
<td>Anomaly Test Rules</td>
<td>Anomaly test rules used for anomaly model testing.</td>
</tr>
<tr>
<td>(sa_metric_anomaly_test_rule)</td>
<td></td>
</tr>
<tr>
<td>Anomaly test results</td>
<td>Results from anomaly model testing, such as bounds and anomaly scores.</td>
</tr>
<tr>
<td>(sa_metric_anomaly_test_result)</td>
<td></td>
</tr>
<tr>
<td>Metric anomaly test config customized parameters</td>
<td>Customized parameters for selected series in anomaly model testing.</td>
</tr>
<tr>
<td>(sa_metric_anomaly_test_config_custom_param)</td>
<td></td>
</tr>
<tr>
<td>Operational Intelligence Properties</td>
<td>Operational Intelligence properties by domain.</td>
</tr>
<tr>
<td>(sa_metric_property)</td>
<td></td>
</tr>
<tr>
<td>Track host name with metric CI</td>
<td>CIs that are bind with node name from the binding event.</td>
</tr>
<tr>
<td>(sa_metric_ci_host)</td>
<td></td>
</tr>
<tr>
<td>Functions supported by metric expression language</td>
<td>Functions supported by metric expression language.</td>
</tr>
<tr>
<td>(sa_metric_expr_function)</td>
<td></td>
</tr>
<tr>
<td>Argument types for functions supported by metric expression language</td>
<td>Argument types for functions supported by metric expression language.</td>
</tr>
<tr>
<td>(sa_metric_expr_func_arg)</td>
<td></td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Track CI Types to metric entries valid for different sources</td>
<td>Contains mappings of CI type to resource class. Used for resource binding during resource lookups given a CI and a <code>resource_path</code> in the event. The <code>sa_ci_type_to_resource_class</code> table contains some default mapping records, and the <code>evt_mgmt_admin</code> role is required to manage this table.</td>
</tr>
<tr>
<td>(sa_metric_ci_type_map)</td>
<td></td>
</tr>
<tr>
<td>CI Type To Resource Class</td>
<td>Parent table used for resource binding.</td>
</tr>
<tr>
<td>(sa_ci_type_to_resource_class)</td>
<td></td>
</tr>
<tr>
<td>CI Resource</td>
<td>Parent table used for resource binding.</td>
</tr>
<tr>
<td>(ci_resource)</td>
<td></td>
</tr>
<tr>
<td>CI Resource Hardware</td>
<td>An extension of the CI Resource (ci_resource) table, used for resource binding.</td>
</tr>
<tr>
<td>(ci_resource_hardware)</td>
<td></td>
</tr>
<tr>
<td>CI Resource Application</td>
<td>An extension of the CI Resource (ci_resource) table, used for resource binding.</td>
</tr>
<tr>
<td>(ci_resource_appl)</td>
<td></td>
</tr>
<tr>
<td>CI Resource Service</td>
<td>An extension of the CI Resource (ci_resource) table, used for resource binding.</td>
</tr>
<tr>
<td>(ci_resource_service)</td>
<td></td>
</tr>
<tr>
<td>CI Resource VM</td>
<td>An extension of the CI Resource (ci_resource) table, used for resource binding.</td>
</tr>
<tr>
<td>(ci_resource_vm_object)</td>
<td></td>
</tr>
<tr>
<td>CI Resource Database</td>
<td>An extension of the CI Resource (ci_resource) table, used for resource binding.</td>
</tr>
<tr>
<td>(ci_resource_database)</td>
<td></td>
</tr>
<tr>
<td>Series Event History</td>
<td>Information about CI s start and end maintenance mode.</td>
</tr>
<tr>
<td>(sa_series_event_history)</td>
<td>Operational Intelligence stores historical information about the times CI s enter and exit maintenance mode if the system property <code>sa.model_learner.maint_event_record_history_enabled</code> is set to true (default). And if the system property <code>sa_metric.maint_exclusion</code> is also set to true (default), metrics from CI s that are in maintenance mode are excluded from model learning.</td>
</tr>
<tr>
<td>Metric Dashboard Metadata</td>
<td>Chart configurations per CI class, for the metrics that are displayed for a CI associated with an alert, in Agent Workspace.</td>
</tr>
<tr>
<td>(sa_metric_dashboard_metadata)</td>
<td></td>
</tr>
<tr>
<td>Metric Config Anomaly Detection Staging</td>
<td>Internal table that is used to help calculate the action level for each entry in the Metric to CI Mapping (sa_metric_map) table.</td>
</tr>
<tr>
<td>(sa_metric_config_anomaly_detection_staging)</td>
<td></td>
</tr>
</tbody>
</table>
### Properties installed

**Note:** To open the System Property (sys_properties) table, enter `sys_properties.list` in the navigation filter. To open the Metric Settings (sa_metric_config_setting) table, enter `sa_metric_config_setting.list` in the navigation filter.

<table>
<thead>
<tr>
<th>Property</th>
<th>Usage</th>
<th>Type</th>
<th>Default value</th>
<th>Location</th>
<th>Learn more</th>
</tr>
</thead>
<tbody>
<tr>
<td>sa.metric.map.with.ci.expiration.sec</td>
<td>Length of time (in seconds) that records for mapping of raw data to CIs remains in effect if the mapping to CI is found.</td>
<td>numeric</td>
<td>432000</td>
<td>System Property (sys_properties) table</td>
<td>Operational Intelligence</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sa.metric.map.without.ci.expiration.sec</td>
<td>Length of time (in seconds) that records for mapping of raw data to CIs remains in effect if the mapping to CI is not found.</td>
<td>numeric</td>
<td>86400</td>
<td>System Property (sys_properties) table</td>
<td>Operational Intelligence</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sa_metric.anomaly_score_stale_interval_in_days</td>
<td>The starting day of the window of time that the Operational Metrics – Process Stale Anomaly Score – Daily scheduled job uses when checking staleness status of anomaly score. The window of time spans from today, until the past number of days specified in this property.</td>
<td>numeric</td>
<td>3</td>
<td>System Property (sys_properties) table</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sa_metric.anomaly_score_next_state_interval_in_days</td>
<td>Number of days by which to extend the window of time used by the Operational Metrics – Process Stale Anomaly Score – Daily scheduled job. Applies if the job runs, but fails to return any data using the window of time specified by the sa_metric.anomaly_score_stale_interval_in_days property.</td>
<td>numeric</td>
<td>7</td>
<td>System Property (sys_properties) table</td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>max_records_per_rest_request</td>
<td>Maximum number of records to download from the instance to the MID Server in each synchronization cycle. This property is for internal use.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: integer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Default value</strong>: 10000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Other possible values</strong>:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Minimum value</strong>: 1000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Maximum value</strong>: 20000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Location</strong>: Metric Settings (sa_metric_config_setting) table.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>normalized_metric_max_queue_size</td>
<td>Size of metrics queue. Anomaly detector reads through this queue and determines for each metric whether it is anomalous. This property is for internal use.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: integer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Default value</strong>: 30000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Other possible values</strong>:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Minimum value</strong>: 10000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Maximum value</strong>: 50000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Location</strong>: Metric Settings (sa_metric_config_setting) table.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>normalized_metric_queue_consumer_count</td>
<td>Number of threads that concurrently perform anomaly detection. This property is for internal use.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: integer</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Default value</strong>: 12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Other possible values</strong>:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Minimum value</strong>: 8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- <strong>Maximum value</strong>: 16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Location</strong>: Metric Settings (sa_metric_config_setting) table.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ignite_data_streamer_thread_count</td>
<td>Number of threads that data streamers use to write data to the cache. The data streamer creates batches from individual data entries and adds these batches to the cache. This property is for internal use.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
|                                | **Type**: integer  
|                                | **Default value**: 8  
|                                | **Other possible values**:  
|                                |   · Minimum value: 8  
|                                |   · Maximum value: 32  
|                                | **Location**: Metric Settings (sa_metric_config_setting) table.                                                                                                                                 |
| ignite_data_streamer_flush_frequency | Frequency of automatic flush of data streamers. The data streamer creates batches from individual data entries and adds these batches to the cache. This property is for internal use. |
|                                | **Type**: integer  
|                                | **Default value**: 1000  
|                                | **Other possible values**:  
|                                |   · Minimum value: 0  
|                                |   · Maximum value: 30000  
|                                | **Location**: Metric Settings (sa_metric_config_setting) table.                                                                                                                                 |
| ignite_data_streamer_buffer_size | Buffer size of data streamers. The data streamer creates batches from individual data entries and adds these batches to the cache. This property is for internal use. |
|                                | **Type**: integer  
|                                | **Default value**: 102400  
|                                | **Other possible values**:  
|                                |   · Minimum value: 1024  
|                                |   · Maximum value: 1024000  
<p>|                                | <strong>Location</strong>: Metric Settings (sa_metric_config_setting) table.                                                                                                                                 |</p>
<table>
<thead>
<tr>
<th>Property</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ignite_data_streamer_batch_size</td>
<td>Size of the batch that is added to a data streamer. The data streamer creates batches from individual data entries and adds these batches to the cache. This property is for internal use.</td>
</tr>
</tbody>
</table>
|                                             | - **Type**: integer  
|                                             | - **Default value**: 5000  
|                                             | - **Other possible values**:  
|                                             |   - Minimum value: 1  
|                                             |   - Maximum value: 10000  
|                                             | - **Location**: Metric Settings (sa_metric_config_setting) table.                                                                                                                                 |
| ignite_data_streamer_batch_flush_frequency   | Frequency of automatic flush of internal batching before adding data to data streamers. The data streamer creates batches from individual data entries and adds these batches to the cache. This property is for internal use. |
|                                             | - **Type**: integer  
|                                             | - **Default value**: 5000  
|                                             | - **Other possible values**:  
|                                             |   - Minimum value: 1000  
|                                             |   - Maximum value: 30000  
|                                             | - **Location**: Metric Settings (sa_metric_config_setting) table.                                                                                                                                 |
| display_monitored_object_type                | Lets you configure Advanced Insights Explorer to show the specific monitored object type that is associated with a referenced source metric type.                                                     |
|                                             | - **Type**: true | false  
|                                             | - **Default value**: false  
|                                             | - **Location**: System Property (sys_properties) table  
|                                             | - **Learn more**: Use Advanced Insights Explorer to query CI metric data                                                                                                                                 |
| sa.metric.use.resource.binding               | Enables resource binding.                                                                                                                                                                             |
|                                             | - **Type**: true | false  
|                                             | - **Default value**: true  
|                                             | - **Location**: System Property (sys_properties) table  
|                                             | - **Learn more**: Metric binding to resources
<table>
<thead>
<tr>
<th>Property</th>
<th>Usage</th>
</tr>
</thead>
</table>
| sa.metric.display.mid.setup.action | Displays the Setup Operational Intelligence related link on the MID Server form, for a MID Server that has been validated and that is running.  
- **Type**: true | false  
- **Default value**: true  
- **Location**: System Property (sys_properties) table  
- **Learn more**: MID Server and MID Server distributed cluster for Operational Intelligence |
| sa.model_learner.maint_event_record_history_enabled | Stores historical information about CIs start and end maintenance mode, in the Series Event History (sa_series_event_history) table.  
- **Type**: true | false  
- **Default value**: true  
- **Location**: System Property (sys_properties) table  
- **Learn more**: Understanding Operational Intelligence |
| sa_metric.maint_exclusion | Excludes metrics for CIs that are in maintenance mode, from model learning.  
- **Type**: true | false  
- **Default value**: true  
- **Location**: System Property (sys_properties) table  
- **Learn more**: Understanding Operational Intelligence |
| sa.model_learner.maint_history_batch_size | Determines the number of data series per batch when storing information about CIs start and end maintenance mode, in the Series Event History (sa_series_event_history) table.  
Adjust the setting of this property if there are related performance issues.  
- **Type**: integer  
- **Default value**: 1000  
- **Location**: System Property (sys_properties) table  
- **Learn more**: Understanding Operational Intelligence |
<table>
<thead>
<tr>
<th>Property</th>
<th>Usage</th>
</tr>
</thead>
</table>
| sa_metric.scheduled_jobs_to_monitor_sys_id_list | Contains the sys_ids of the 3 scheduled jobs that self-health monitoring for Operational Intelligence monitors:  
• Operational Intelligence — Metric configuration job  
• Operational Intelligence — Sync tables with mid  
• Operational Intelligence — Validate MID Distributed cluster |

### Create or configure data source registration

Operational Intelligence uses various data sources to collect raw metric data which is later processed and analyzed. These data sources must be registered in the system. For example, you can configure the pre-defined SCOM data source registration to control some aspects of how it processes raw data, or register a new data source.

Role required: evt_mgmt_admin

If raw data is pushed or pulled from a data source that is not registered, Operational Intelligence automatically registers that source and adds a record in the SA Metric Type Registration table. By default, Registration Mode is set to Active, Type Default Mode is set to Active, and Generate Missing CIs is set to false in the new record.

1. Navigate to **Operational Intelligence > Configure > Data Source Registration**. All existing registered data sources are displayed.
2. Click an existing data source to edit, or click **New** to create a new one.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Source of metric data.</td>
</tr>
</tbody>
</table>
| Registration Mode | Determines how to process an unknown metric type for the data source that is detected in raw data:  
• **Active**: A new metric is automatically generated based on the unknown metric.  
• **Inactive**: The unknown metric is discarded.  
For SCOM, value is set to **Inactive** by default. |
| Type default mode | Determines if a metric that is automatically generated by the system, is set to be active or inactive.  
For SCOM, value is set to **Inactive** by default. |
Choose and configure metrics to monitor

Operational Intelligence uses data sources that can be monitoring hundreds of metrics for all CIs. Choose for each data source type which details are important for which CIs, and then activate or deactivate the respective monitor type to control the amount of data that is being processed.

See [Get started with Operational Intelligence](#) for the high level steps of setting up Operational Intelligence.

Role required: evt_mgmt_admin

The SCOM data source, available by default, collects about 400 metric types from discovered devices such as '% Available Memory' and '% of Free Space'. Processing data from all metric types can overload the system and impact performance. It might be necessary to deactivate selected metric types for selected CIs to reduce that load.

The overall number of metrics that can be processed, depends on the number of MID Servers and their hardware configuration. For details about hardware requirements and metrics scalability, see [MID Server and MID Server distributed cluster for Operational Intelligence](#).

**Note:** The Active setting (true or false) for a metric in the Monitoring System Metric Types (sa_source_metric_type) table takes precedence over the setting for the corresponding metric in the Metric To CI Mappings (sa_metric_map) table. If a metric type in the Monitoring System Metric Types (sa_source_metric_type) table is disabled, all records related to the corresponding metrics are removed from the Metric To CI Mappings (sa_metric_map) table.

1. Navigate to Operational Intelligence > Configure > Metric Types.
   The Monitoring System Metric Types displays the metric types that are being collected per data source, per CI type.
2. Review the Monitoring System Metric Types list, and set the value in the Active column to false or true as necessary.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Data source monitoring the metric type.</td>
</tr>
<tr>
<td>Source Metric Type</td>
<td>Metric such as '% Free Space' that is being monitored for the CI type.</td>
</tr>
<tr>
<td>Monitored Object Type</td>
<td>CI type that is being monitored.</td>
</tr>
<tr>
<td>Active</td>
<td>Determines whether the metric data is processed by Event Management alert aggregation and RCA metrics.</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Unit</td>
<td>Unit type that is associated with the source metric type, displayed in the Operational Intelligence reports.</td>
</tr>
</tbody>
</table>

**Metric binding**

After metric data is collected, Operational Intelligence identifies the CIs and the resources to bind the data to.

The data that is collected on the MID Server is raw and does not relate to any specific CI or resource in the CMDB. To be useful, the data goes through a normalization process that uses CMDB identification rules and event rules to uniquely identify CIs, and to map and bind them to the raw data. Raw data can also be mapped and bound to resources, if binding to a CI is successful and if resource binding is configured.

Records for mapping raw data to CIs are automatically generated and remain in effect for a specified length of time determined by the properties:

- **sa.metric.map.with.ci.expiration.sec**: If the mapping to the CI was found. Set by default to be valid for five days.
- **sa.metric.map.without.ci.expiration.sec**: If mapping to the CI was not found. Set by default to be valid for 24 hours.

When similar metric data arrives within that time period, the existing mapping is used to match the data to CIs. At the end of the time period, metric-to-CI records expire. Also, a change in the event rules triggers an immediate expiration of the respective metric-to-CI records. Next time that raw metric data arrives, it will be normalized again. When Discovery adds or removes CIs, mappings are adjusted to reflect these changes at the next cycle.

**Metric binding to resources**

Bind metrics to resources to simplify metric events binding by enabling binding to resources such as specific disks or web pages, in addition to binding to CIs.

Operational Intelligence models metric series in either of the following methods:

- Binding a metric series to a CI and to the metric being monitored for that CI using a 'CI/Metric' format. For example, 'win_server_001/response_time_mean'.
- Binding a metric series to a CI, a resource within that CI, and the metric being monitored for that resource, using a 'CI/Resource/Metric' format. For example, 'win_server_001/Disk — C/disk_usage'. This method is referred to as 'resource binding'.

The first method enables modeling, storing, aggregating, and querying data at the CI level. However, resource binding creates resource records for specific monitored entities, such as an individual web page or a disk. Therefore, resource binding is more efficient when there are multiple resources of similar types within a CI, and metrics applicable to a category of these resources are mapped to these resources.

Use cases:

- Common examples are disks, processors, and network interfaces. In cases in which these entities are being monitored but do not exist in the CMDB, using resources for metric binding is useful.
- Some monitoring solutions capture metric data within services such as KPIs for individual web pages in an application. In such cases where the entity being monitored is not a configuration item, metric binding to resources can be helpful.
In those situations, using resource binding results in more meaningful aggregations across similar metrics (such as: avg disk_usage for a host across all disks).

**Resource binding process**

The MID Server generates metric binding events that are processed by the instance. When processing a metric binding event, there is an attempt to identify the CI to which this metric binding event belongs. If this attempt is successful, then the metric binding event is bound to the identified CI and binding to a resource is attempted. The following steps describe the attempt to bind a metric binding event to a resource:

1. Identify the CI class of the CI that was bound to the metric binding event.
2. Locate the resource class which is mapped to that CI class (using the CI Type To Resource Class \(sa\_ci\_type\_to\_resource\_class\) table).
3. Read the **resource_path** attribute value in the **additional_information** field in the metric binding event.
4. Check if a resource record exists in the resource class table, in which **name** is equal to **resource_path** and **cmdb\_ci** is equal to the CI that was bound to the metric binding event.
5. If such resource record exists, then the metric binding event is bound to that resource. Otherwise, a new resource record is created with the preceding values and the metric binding event is bound to the newly created resource.

**Configure resource binding**

- Ensure that the **sa\_metric\_use\_resource\_binding** system property is set to true (default).
- For series intended to be bound to resources, ensure that the **resource_path** attribute is populated by doing either step:
  - Use an event rule to add the **resource_path** attribute to the **Additional information** field in events. For more information, see [Create an event rule to map metrics to specific CI](#).
  - Populate the **resource_path** attribute as a part of the respective CI identifier when data is sent to the MID Server for processing.
- Review the default mappings in the CI Type To Resource Class \(sa\_ci\_type\_to\_resource\_class\) table and adjust as needed. It is critical that mappings are set as desired prior to data processing.

  Managing the CI Type To Resource Class table requires the evt_mgmt_admin role.

**Resource tables**

Resource binding uses an underlying hierarchy structure of resources which is a subset of the (cmdb\_ci) hierarchy. Operational Intelligence installs the following resource tables:

- CI Resource (ci\_resource) (parent table):
- Tables that extend CI Resource (ci\_resource):
  - ci\_resource\_hardware
  - ci\_resource\_appl
  - ci\_resource\_service
  - ci\_resource\_vm\_object
Mapping CIs to resources

Mappings of CI classes to resource classes are stored in the CI Type To Resource Class (sa_ci_type_to_resource_class) table. This table is installed with Operational Intelligence, and is used during metric binding to resources.

<table>
<thead>
<tr>
<th>CI class</th>
<th>Resource class</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdb_ci_hardware</td>
<td>ci_resource_hardware</td>
</tr>
<tr>
<td>cmdb_ci_appl</td>
<td>ci_resource_appl</td>
</tr>
<tr>
<td>cmdb_ci_service</td>
<td>ci_resource_service</td>
</tr>
<tr>
<td>cmdb_ci_database</td>
<td>ci_resource_database</td>
</tr>
<tr>
<td>cmdb_ci_vm_object</td>
<td>ci_resource_vm_object</td>
</tr>
</tbody>
</table>

Create an event rule to map metrics to specific CI

Create event rules to map incoming raw metric data to specific CIs, to optionally modify metric names, and to populate the resource_path attribute for resource binding.

The system generates events for every new raw metric data for which there are no CI binding specifications. You can create event rules that filter those events to map the incoming metric data to specific CIs. In such event rules, you can for example define a filter that is based on the monitoring object (such as SCOM).

To use resource binding, the resource_path attribute of events must be populated. You can configure an event rule to ensure that the resource_path attribute is included in the Additional information field in events. For more information about configuring and using resource binding, see resource binding.

For example, in the Event Rule designer you can update the event rule SCOM IIS Web Server as follows:

- On the Event Filter tab, define filter to match only IIS Web Servers.
- On the Binding tab, set Class to Microsoft IIS Web Server.

Create an event rule to bind metric events to host CIs

Create event rules to explicitly allow binding of metric events to host CIs.

Role required: evt_mgmt_admin

Starting with the London release, Operational Intelligence no longer relies on the default Event Management binding workflow which binds nodes to host CIs. Event Management is configured to apply the default host binding algorithm when an event rule cannot find a CI to bind to, based on event data. In Operational Intelligence, with a given event rule and two different events, there can be a situation where one event binds to the correct CI and another event cannot bind. In this case, the same metric types would be created for the selected CI and for the host level CI, causing confusion.

For example: With an event rule that binds the disk percentage metric to a storage disk, one event can find a disk CI and another event cannot. This results in the disk percentage metric being created for a disk CI and for a host CI.
To avoid this confusion, Event Management default host binding is disabled for metric events in the base system. To allow host binding for metric events, explicitly create the following event rule in the Event Management event rule designer. The underlying binding functionality of the event rule designer is used to map metrics to CIs.

1. Navigate to Event Management > Rules > Event Rules to open the event rule designer.
2. Click New and ensure that the rule is set to Active.
3. On the Event Rule Info tab:
   a) Set Source to the data source that this rule applies to.
   b) Enter a Description such as Bind metrics at host level.
4. On the Event Filter tab, set All of these conditions must be met to Classification is Metric.
5. Click Save.

Disable a metric for a CI

Operational Intelligence can be configured to collect significant amounts of data, some of which might not be necessary. To improve performance, you can disable a specific metric for a specific CI to stop processing data related to the specified metric and CI.

Role required: evt_mgmt_admin

![Note: The Active setting (true or false) for a source metric in the Monitoring System Metric Types (sa_source_metric_type) table takes precedence over the setting for the corresponding metric in the Metric To CI Mappings (sa_metric_map) table. If a source metric type in the Monitoring System Metric Types (sa_source_metric_type) table is disabled, all records related to the corresponding metrics are removed from the Metric To CI Mappings (sa_metric_map) table.](image)

1. Navigate to Operational Intelligence > Configure > Metric to CI.
2. Locate the record for the metric/CI that you want to disable, and set its Active column to false.

Disable event collection

For performance reasons, you might want to disable the collection of events from a data source from which metric data is also collected. You can disable event collection for data sources such as Nagios XI server, SolarWinds monitoring system, and Zabbix server.

Role required: evt_mgmt_admin

![Note: To restart event collection after it has been disabled, you will need to restore the connector's original script.](image)

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Select a connector instance for the data source for which to disable event collection. For example, select a connector instance for Nagios, SolarWinds, or Zabbix.
3. Set Event collection schedule (seconds) to a large value such as 20,000,000. Such setting avoids unnecessary communication with the MID Server, optimizing performance.
4. Click Update.
5. Navigate to **MID Server > Script Includes** and extend the connector's script (such as **NagiosMetrics_JS**, **SolarWindsJS**, or **ZabbixJS**) by overriding the **execute** method. Add a code snippet such as the following, and then click **Update**:

**Snippet for Nagios**

```javascript
var NagiosJSCustom = Class.create();

// Extending Nagios default script
NagiosJSCustom.prototype = Object.extendsObject(NagiosJS, {
    // Override execute method
    execute: function() {
        var retVal = {};
        retVal['status'] = SUCCESS.toString();
        retVal['error_message'] = "Disabled by custom Class, 'Last event collection status' will be 'error' and this is normal. To achieve success status, enable event pull in custom class(NagiosJSCustom)";
        return retVal;
    };
});
```

For the **SolarWinds** connector use:

```javascript
var SolarWindsJSCustom = Class.create();

// Extending solarwinds default script
SolarWindsJSCustom.prototype = Object.extendsObject(SolarWindsJS, {
//
});
```

For the **Zabbix** connector use:

```javascript
var ZabbixJSCustom = Class.create();

// Extending zabbix default script
ZabbixJSCustom.prototype = Object.extendsObject(ZabbixJS, {
//
});
```

### Configure anomaly score thresholds

Operational Intelligence scores anomalies on a range 0-10. This range is broken down to the five levels of event severities, each represented by a different color in the Insights Explorer and in the Anomaly Map. You can configure the anomaly score threshold for each level of severity.

Role required: **evt_mgmt_admin**

Anomaly score thresholds are defined in the table **Anomaly Score to Event Severity Map** (`sa_metric_anomaly_score_to_event_severity_map`). For example, by default, anomaly scores from 8 through 9 are severity 2 (Major), and are displayed by dark orange color in the Insights Explorer and in the Anomaly Map.

1. Navigate to **Operational Intelligence > Settings > Anomaly Score to Severity Mapping**.
2. In the **Metric Anomaly Score to Event Severity Maps** list view, double-click the event severity that you want to modify.
3. Modify the range of anomaly scores that map to the respective event severity.
4. Click **Update**.
Configure metric connector instances

Use Operational Intelligence connector definitions to create connector instances that pull metric data from event collection systems. Operational Intelligence then analyses metric data to detect anomalies and provide statistical scores.

Operational Intelligence plugin (com.snc.sa.metric) must be activated for the Operational Intelligence connectors to appear in the base system. In addition, each connector requires a credential that lets the instance access the data collection system accounts. You can use an existing credential or create a new one as needed.

Event Management and Operational Intelligence share some connector definitions and connector instances when pulling events and metric data from the same data source. Therefore, if a connector instance for a specific data source already exists, configure that connector instance to also pull metric data instead of creating another connector instance. Create a new connector instance for a data source only if such connector instance does not exist.

Supported connectors specific to Operational Intelligence

<table>
<thead>
<tr>
<th>Connector to</th>
<th>Supported versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Web Service (AWS)</td>
<td>NA</td>
</tr>
<tr>
<td>Microsoft Azure platform</td>
<td>NA</td>
</tr>
</tbody>
</table>

Supported connectors shared between Operational Intelligence and Event Management

<table>
<thead>
<tr>
<th>Connector to</th>
<th>Supported versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nagios XI server</td>
<td>5.4.0</td>
</tr>
<tr>
<td>SolarWinds monitor</td>
<td>NPM 12.0, SAM 6.2.1</td>
</tr>
<tr>
<td>Zabbix server</td>
<td>3.0, 3.2</td>
</tr>
</tbody>
</table>

Create an Azure metric connector instance

Operational Intelligence provides the Azure connector definition which you can use to create a connector instance to pull metric data from the Microsoft Azure platform. Operational Intelligence then analyses metric data to provide statistical scores and detect anomalies.

Operational Intelligence plugin (com.snc.sa.metric) must be activated for the Operational Intelligence connectors to appear in the base system. In addition, the Azure connector requires an Azure Service Principal type credential that lets the instance access Azure platform accounts. You can use an existing credential or create a new one.

Role required: evt_mgmt_admin

After you configure the Azure connector, you can run a test to ensure that the MID Server can authenticate itself to the Azure platform account.

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click **New**.
3. Fill out the Connector Instance form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the connector instance record, such as the name of event source host.</td>
</tr>
<tr>
<td>Description</td>
<td>Any optional information that the administrator wants to use to identify this record.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Vendor and protocol used to gather metric data from the external source. Select <strong>AzureConnector</strong>.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Not used for this connector. Therefore any value such as 1.1.1.1, is valid.</td>
</tr>
<tr>
<td>Metrics collection</td>
<td>Designates collection of metric data for Operational Intelligence.</td>
</tr>
<tr>
<td>Metrics collection last run time</td>
<td>The date and time of the most recent metric data collection. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last metrics collection status</td>
<td>Status of the metric data collection activity. The value of this field is automatically populated.</td>
</tr>
<tr>
<td></td>
<td>This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and if the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics collection schedule (seconds)</td>
<td>The time, in seconds, to repeat the metric data collection scheduled job.</td>
</tr>
<tr>
<td></td>
<td>This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Credential</td>
<td>A valid credential of the Azure Service Principal type, to the Microsoft Azure platform data source host. Click the <strong>Lookup using list</strong> icon to select a credential, or to create a new one.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message received by the connector. If the test connector fails, an error message appears in this field.</td>
</tr>
</tbody>
</table>

4. In the MID Servers for Connectors related list, click **Insert a new row** and then select an Operational Intelligence MID Server to send metric data to.
5. Right-click the form header and select **Save**.
6. In the Connector Instance Values related list, for each name, double-click the **Value** column and enter values as follows:
<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>max_fetch_interval_min</td>
<td>Maximum time in minutes to go back to for retrieving data. Example value: 180</td>
</tr>
<tr>
<td>offset_min</td>
<td>Number of minutes to shift the time interval back by. Used to offset a delay in the availability of data on external systems. For example, data for 12PM is available only at 12:00:30 which is a 30 seconds delay. offset_min ensures reading that data, which otherwise might have been missed. Example value: 5</td>
</tr>
<tr>
<td>subscription_id</td>
<td>A unique GUID for your subscription to Azure services. To obtain your subscription ID: 1. Log on to the Azure portal. 2. In the left navigation panel, click Subscriptions to list your subscriptions and the subscription ID.</td>
</tr>
</tbody>
</table>

7. Click Test connector to verify that the MID Server can communicate with the external server host.
   
   If the test fails, follow the instructions on the page to correct the problem and then run another test. The description of the reason for failure is shown in the MID Server log, according to the type of error, such as Events, Metrics, or Bi-Directional.

8. After a successful test, select the Active check box and then click Update.

Create an AWS metric connector instance

Operational Intelligence provides the AWS CloudWatch connector definition which you can use to create a connector instance to pull metric data from Amazon Web Service (AWS). Operational Intelligence then analyses metric data to provide statistical scores and detect anomalies.

Operational Intelligence plugin (com.snc.sa.metric) must be activated for the Operational Intelligence connectors to appear in the base system. In addition, the AWS connector requires an AWS Credentials type credential that lets the instance access AWS accounts. You can use an existing credential, or create a new one.

Role required: evt_mgmt_admin

After you configure the AWS CloudWatch connector, you can run a test to ensure that the MID Server can authenticate itself to the AWS account and read data from the account.

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.
2. Click New.
3. Fill out the Connector Instance form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the connector instance record, such as the name of event source host.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>Any optional information that the administrator wants to use to identify this record.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Vendor and protocol used to gather metric data from the external source.</td>
</tr>
<tr>
<td></td>
<td>Select AWS CloudWatch.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Not used for this connector. Therefore any value such as 1.1.1.1, is valid.</td>
</tr>
<tr>
<td>Metrics collection</td>
<td>Designates collection of metric data for Operational Intelligence.</td>
</tr>
<tr>
<td>Metrics collection last run time</td>
<td>The date and time of the most recent metric data collection. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last metrics collection status</td>
<td>Status of the metric data collection activity. The value of this field is automatically populated. This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and if the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics collection schedule (seconds)</td>
<td>The time, in seconds, to repeat the metric data collection scheduled job.</td>
</tr>
<tr>
<td></td>
<td>This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Credential</td>
<td>A valid credential of the AWS Credentials type, to the AWS metric data source host. Click the <strong>Lookup using list</strong> icon to select a credential, or to create a new one.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message received by the connector. If the test connector fails, an error message appears in this field.</td>
</tr>
</tbody>
</table>

4. In the MID Servers for Connectors related list, click **Insert a new row** and then select an Operational Intelligence MID Server to send metric data to.

5. Right-click the form header and select **Save**. The Connector Instance Values related list appears.

6. In the Connector Instance Values related list, for each name, double-click the **Value** column and enter values as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>account_id</td>
<td>The AWS account identification.</td>
</tr>
<tr>
<td>max_fetch_interval_min</td>
<td>Maximum time in minutes to go back to for retrieving data. Example value: 180</td>
</tr>
</tbody>
</table>
### Offset Min

**Name:** offset_min  
**Value:** Number of minutes to shift the time interval back by.  
**Description:** Used to offset a delay in the availability of data on external systems. For example, data for 12PM is available only at 12:00:30 which is a 30 seconds delay. `offset_min` ensures reading that data, which otherwise might have been missed.  
**Example value:** 5

### Period Sec

**Name:** period_sec  
**Value:** The granularity in seconds of the returned data points.  
**Example value:** 60

### Regions

**Name:** regions  
**Value:** Regions (as they are defined in the AWS account), to get data from.

---

7. Click **Test connector** to verify that the MID Server can communicate with the external server host.  
   If the test fails, follow the instructions on the page to correct the problem and then run another test. The description of the reason for failure is shown in the MID Server log, according to the type of error, such as Events, Metrics, or Bi-Directional.  
8. After a successful test, select the **Active** check box and then click **Update**.

---

### Configure a Zabbix metric connector instance

Event Management provides the Zabbix connector definition which you can use to configure or create a connector instance to pull metric data from a Zabbix server. Operational Intelligence then analyses metric data to provide statistical scores and detect anomalies.

Event Management and Operational Intelligence share a single connector definition and a single connector instance to pull events and metric data from a Zabbix server. You can configure this shared connector to pull from the data source only events, only metric data, or both. Therefore, it is important to search all connector instances for an existing Zabbix connector instance that is already configured for pulling events. If such connector exists, update that connector instance to also pull metric data.

To start:

1. Decide if you want to create a Zabbix connector instance for pulling only events, only metric data, or both.  
2. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances** and search for an existing Zabbix connector instance that is already configured for pulling events.  
3. If you want to create a Zabbix connector instance for pulling both events and metric data and a Zabbix connector does not exist:
   1. **Configure event collection from Zabbix server.**  
   2. **Configure** that same connector to pull metric data as described in the following procedure.

The Zabbix connector instance requires a credential that lets the instance access Zabbix monitoring system accounts. You can use an existing credential or **create a new one**.
Role required: evt_mgmt_admin

After you configure the Zabbix connector, you can run a test to ensure that the MID Server can authenticate itself to the Zabbix server account and read data from the account.

**Note:** Depending on whether you are creating a new connector instance or updating an existing one, some of the following steps might not be necessary. Do not skip the *Optional* steps if you are creating a new connector instance for pulling metric data only.

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances**.
2. Select an existing Zabbix connector instance that is already configured to pull events, or click **New**.
3. Fill out the Connector Instance form.
   - If you are updating a connector instance that is already configured for Event Management, then some of the following fields are already set.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the connector instance record, such as the name of event source host.</td>
</tr>
<tr>
<td>Description</td>
<td>Any optional information that the administrator wants to use to identify this record.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Vendor and protocol used to gather metric data from the external source. Select Zabbix.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Not used for this connector. Therefore any value such as 1.1.1.1, is valid.</td>
</tr>
<tr>
<td>Metrics collection</td>
<td>Designates collection of metric data for Operational Intelligence.</td>
</tr>
<tr>
<td>Metrics collection last run time</td>
<td>The date and time of the most recent metric data collection. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last metrics collection status</td>
<td>Status of the metric data collection activity. The value of this field is automatically populated. This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and if the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Metrics collection schedule (seconds)</td>
<td>The time, in seconds, to repeat the metric data collection scheduled job. This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and when the <strong>Metrics collection</strong> option is selected.</td>
</tr>
<tr>
<td>Credential</td>
<td>A valid credential of the Basic Auth Credentials type, to the Zabbix metric data source host. Click the <strong>Lookup using list</strong> icon to select a credential, or to create a new one.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message received by the connector. If the test connector fails, an error message appears in this field.</td>
</tr>
</tbody>
</table>

4. Optional: In the MID Servers for Connectors related list, click **Insert a new row** and then select an Operational Intelligence MID Server to send metric data to.

5. Optional: Right-click the form header and select **Save**. The Connector Instance Values related list appears.

6. In the Connector Instance Values related list, review the following parameters, each affecting event collection, or metric data collection, or both. These parameters are preconfigured and it is not necessary to modify the default values. However, it is possible to modify a value by double-clicking the **Value** column and entering a new value.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>groupids</td>
<td>One or more host group IDs from which to collect metrics. To specify multiple host IDs, enter a comma-separated list of host group IDs such as 1, 2, 7. To identify a host group ID 1. On Zabbix server, navigate to <strong>Configuration &gt; Host Groups</strong>. 2. Select a group and note the group’s ID.</td>
</tr>
<tr>
<td>max_fetch_interval_min</td>
<td>Maximum time in minutes to go back to for retrieving data. Optional value: 180</td>
</tr>
<tr>
<td>offset_min</td>
<td>Number of minutes to shift the time interval back by. Used to offset a delay in the availability of data on external systems. For example, data for 12PM is available only at 12:00:30 which is a 30 seconds delay. <strong>offset_min</strong> ensures reading that data, which otherwise might have been missed. Optional value: 5</td>
</tr>
<tr>
<td>port</td>
<td>The port number to use with the specified <strong>protocol</strong>. For example, <strong>protocol</strong> is set to <strong>http</strong> and <strong>port</strong> is set to 80.</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol that is used with Zabbix server, such as <strong>http</strong> or <strong>https</strong>.</td>
</tr>
<tr>
<td>with_triggers</td>
<td>If <strong>true</strong>, then the connector collects only metrics that have triggers. With any other value, the connector collects data for all metrics.</td>
</tr>
</tbody>
</table>

7. Click **Test connector** to verify that the MID Server can authenticate itself to the Zabbix server account to enable communication. If the test fails, follow the instructions on the page to correct the problem and then run another test. The description of the reason for failure is
shown in the MID Server log, according to the type of error, such as Events, Metrics, or Bi-
Directional.

8. After a successful test, select the **Active** check box and then click **Update**.

9. Optional: **Disable event collection**.

1. **Create or configure data source registration** to register new metric types incoming from the Zabbix server.

2. **Enable or disable metrics** to limit metric collection from the Zabbix server to only specific data that is needed.

3. To ensure that metric data is pulled:
   - Verify that **Collect metrics** in the connector definition is selected.
   - Verify that **Metrics collection** in the connector instance is selected.
   - Verify that **Registration mode** in the metric type registration is set to **Active**.

**Configure a SolarWinds metric connector instance**

Event Management provides the SolarWinds connector definition which you can use to configure or create a connector instance to pull metric data from the SolarWinds monitoring system. Operational Intelligence then analyses metric data to provide statistical scores and detect anomalies.

Event Management and Operational Intelligence share a single connector definition and a single connector instance to pull events and metric data from the SolarWinds data source. You can configure this shared connector to pull from the data source only events, only metric data, or both.

Therefore, it is important to search all connector instances for an existing SolarWinds connector instance that is already configured for pulling events. If such connector exists, update that connector instance to also pull metric data.

To start:

1. Decide if you want to create a SolarWinds connector instance for pulling only events, only metric data, or both.

2. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances** and search for an existing SolarWinds connector instance that is already configured for pulling events.

3. If you want to create a SolarWinds connector instance for pulling both events and metric data and a SolarWinds connector does not exist:
   a. **Create a SolarWinds connector instance for pulling events**.
   b. Configure that same connector to pull metric data as described in the following procedure.

The SolarWinds connector instance requires a credential that lets the instance access SolarWinds monitoring system accounts. You can use an existing credential or create a new one.

Role required: evt_mgmt_admin

The SolarWinds connector does not pull all the metric data that the SolarWinds monitoring system collects, only the following metrics from the following tables:

- CPU and memory metrics from table: (Orion.CPULoad)
  - MinLoad, MaxLoad, AvgLoad, MinMemoryUsed, MaxMemoryUsed, AvgMemoryUsed, AvgPercentMemoryUsed
Default event rules bind CPU metrics at the Host level CIs.

- Interface traffic metrics from table: (Orion.NPM.InterfaceTraffic):

Default event rules bind interface metrics at the Interface level CIs.

- Response time metrics from table: (Orion.ResponseTime):
  - AvgResponseTime, MinResponseTime, MaxResponseTime, PercentLoss, PercentDown, Availability

Default event rules bind response time metrics at the Host level CIs.

- Volume usage metrics from table: (Orion.VolumeUsageHistory):
  - DiskSize, AvgDiskUsed, MinDiskUsed, MaxDiskUsed, PercentDiskUsed, AllocationFailures

Default event rules bind volume performance metrics at the Volume level CIs.

- Volume performance metrics from table: (Orion.VolumePerformanceHistory):
  - AvgDiskQueueLength, MinDiskQueueLength, MaxDiskQueueLength, AvgDiskTransfer, MinDiskTransfer, MaxDiskTransfer, AvgDiskReads, MinDiskReads, MaxDiskReads, AvgDiskWrites, MinDiskWrites, MaxDiskWrites

Default event rules bind volume usage metrics at the Volume level CIs.

---

**Note:** Depending on whether you are creating a new connector instance or updating an existing one, some of the following steps might not be necessary. Do not skip the **Optional** steps if you are creating a new connector instance for pulling metric data only.

1. Navigate to **Event Management > Event Connectors (Pull) > Connector Instances**.
2. Select an existing SolarWinds connector instance that is already configured to pull events, or click **New**.
3. Fill out the connector instance form.

   If you are updating a connector instance that is already configured for Event Management, then some of the following fields are already set.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the connector instance record, such as the name of event source host.</td>
</tr>
<tr>
<td>Description</td>
<td>Any optional information that the administrator wants to use to identify this record.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Vendor and protocol used to gather metric data from the external source. Select SolarWinds.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Host IP of the SolarWinds server.</td>
</tr>
<tr>
<td>Credential</td>
<td>A valid credential to the SolarWinds host. Click the <strong>Lookup using list</strong> icon to select a credential, or to create a new one.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Metrics collection</td>
<td>Designates collection of metric data for Operational Intelligence.</td>
</tr>
<tr>
<td>Metrics collection last run time</td>
<td>The date and time of the most recent metric data collection. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last metrics collection status</td>
<td>Status of the metric data collection activity. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Metrics collection schedule (seconds)</td>
<td>The time, in seconds, to repeat the metric data collection scheduled job.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message received by the connector. If the test connector fails, an error message appears in this field.</td>
</tr>
</tbody>
</table>

4. Optional: In the MID Servers for Connectors related list, click **Insert a new row** and then select an Operational Intelligence MID Server to send metric data to.

5. Optional: Right-click the form header and select **Save**. The Connector Instance Values related list appears.

6. In the Connector Instance Values related list, review the following parameters, each affecting event collection, or metric data collection, or both. These parameters are preconfigured and it is not necessary to modify the default values. However, it is possible to modify a value by double-clicking the **Value** column and entering a new value.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial_sync_in_days</td>
<td>Number of days of data that the initial pull must contain. Default value is 7 days.</td>
</tr>
<tr>
<td>max_fetch_interval_min</td>
<td>Maximum time in minutes to go back to for retrieving data. Optional value: 180</td>
</tr>
<tr>
<td>nodes_custom_properties</td>
<td>SolarWind property names that are associated with the relevant node. This information is added to the Additional Information field of the event in JSON format. SolarWinds monitoring system custom properties are user-defined fields, such as country, that you can associate with monitored network objects. Separate multiple entries with a comma.</td>
</tr>
<tr>
<td>offset_min</td>
<td>Number of minutes to shift the time interval back by. Used to offset a delay in the availability of data on external systems. For example, data for 12PM is available only at 12:00:30 which is a 30 seconds delay. offset_min ensures reading that data, which otherwise might have been missed. Optional min value: greater than 1</td>
</tr>
</tbody>
</table>
7. Click **Test connector** to verify that the MID Server can authenticate itself to the SolarWinds server account to enable communication. If the test fails, follow the instructions on the page to correct the problem and then run another test. The description of the reason for failure is shown in the MID Server log, according to the type of error, such as Events, Metrics, or Bi-Directional.

8. After a successful test, select the **Active** check box and then click **Update**.

9. Optional: **Disable event collection**.

1. **Create or configure data source registration** to register new metric types incoming from the SolarWinds monitoring system.

2. **Enable or disable metrics** to limit metric collection from the SolarWinds monitoring system to only specific data that is needed.

3. To ensure that metric data is pulled:
   - Verify that **Collect metrics** in the connector definition is selected.
   - Verify that **Metrics collection** in the connector instance is selected.
   - Verify that **Registration mode** in the metric type registration is set to **Active**.

**Add optional parameters to the connector definition**
You can add optional parameters to the SolarWinds connector definition, which then appear as optional values for the instance.

Role required: admin

You can add the following connector parameters:

<table>
<thead>
<tr>
<th>Connector parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>debug</td>
<td>A basic level of logging for the queries and responses between the MID Server and the SolarWinds system.</td>
</tr>
<tr>
<td>debug_response</td>
<td>A more detailed level of logging for the queries and responses between the MID Server and the SolarWinds system. Add this parameter temporarily to help with troubleshooting.</td>
</tr>
</tbody>
</table>

1. Navigate to **Event Management > Connector Instances** and edit each SolarWinds connector instance:
   a) Clear **Active**.
   b) Click **Update**.


3. At the bottom of the Connector Parameters list, click **Insert a new row** and add a parameter.

4. Click **Update**.

5. Navigate back to **Event Management > Connector Instances** and edit each SolarWinds connector instance:
a) In the Connector Instance Values section on the Connector Instance form, set values for the new parameters.

b) Select Active.

c) Click Update.

Configure Nagios metric connector instance

Event Management provides the Nagios connector definition which you can use to configure or create a connector instance to pull metric data from the Nagios XI server. Operational Intelligence then analyses metric data to provide statistical scores and detect anomalies.

Event Management and Operational Intelligence share a single connector definition and a single connector instance to pull events and metric data from the Nagios XI data source. You can configure this shared connector to pull from the data source only events, or events and metric data.

Therefore, it is important to search all connector instances for an existing Nagios connector instance that is already configured for pulling events. If such connector exists, you will update that connector instance to also pull metric data.

To start:

1. Decide if you want to create a Nagios connector instance for pulling only events, event and metric data.

2. Navigate to Event Management > Event Connectors (Pull) > Connector Instances and search for an existing Nagios connector instance that is already configured for pulling events.

3. If you want to create a Nagios connector instance for pulling both events and metric data and a Nagios connector does not exist:
   a. Configure event collection from Nagios XI.
   b. Configure that same connector to pull metric data as described in the following procedure.

The Nagios connector instance requires a credential that lets the instance access Nagios XI server accounts. You can use an existing credential or create a new one.

Role required: evt_mgmt_admin

![Note: Depending on whether you are creating a new connector instance or updating an existing one, some of the following steps might not be necessary. Do not skip the Optional steps if you are creating a new connector instance for pulling metric data only.]

1. Navigate to Event Management > Event Connectors (Pull) > Connector Instances.

2. Select an existing Nagios connector instance that is already configured to pull events, or click New.

3. Fill out the connector instance form.
   If you are updating a connector instance that is already configured for Event Management, then some of the following fields are already set.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A unique name for the connector instance record, such as the name of event source host.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>Any optional information that the administrator wants to use to identify this record.</td>
</tr>
<tr>
<td>Connector definition</td>
<td>Vendor and protocol used to gather metric data from the external source. Select NagiosXI.</td>
</tr>
<tr>
<td>Host IP</td>
<td>Host IP of the Nagios server.</td>
</tr>
<tr>
<td>Credential</td>
<td>A valid credential to the Nagios data source host. Click the Lookup using list icon to select a credential, or to create a new one.</td>
</tr>
<tr>
<td>Metrics collection</td>
<td>Designates collection of metric data for Operational Intelligence.</td>
</tr>
<tr>
<td>Metrics collection last run time</td>
<td>The date and time of the most recent metric data collection. The value of this field is automatically populated.</td>
</tr>
<tr>
<td>Last metrics collection status</td>
<td>Status of the metric data collection activity. The value of this field is automatically populated. This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and if the Metrics collection option is selected.</td>
</tr>
<tr>
<td>Metrics collection schedule (seconds)</td>
<td>The time, in seconds, to repeat the metric data collection scheduled job. This option appears only if the connector definition that is selected supports metric collection (such as SCOM) and when the Metrics collection option is selected.</td>
</tr>
<tr>
<td>Metrics database host</td>
<td>IP address or the host name of the database used by the Nagios system.</td>
</tr>
<tr>
<td>Metrics database port</td>
<td>Port for the database used by the Nagios system.</td>
</tr>
<tr>
<td>Metrics database credential</td>
<td>Credentials for the database that is used by the Nagios system.</td>
</tr>
<tr>
<td>Last error message</td>
<td>Last error message received by the connector. If the test connector fails, an error message appears in this field.</td>
</tr>
</tbody>
</table>

4. Optional: In the MID Servers for Connectors related list, click **Insert a new row** and then select an Operational Intelligence MID Server to send metric data to.

5. Optional: Right-click the form header and select **Save**. The Connector Instance Values related list appears.

6. In the Connector Instance Values related list, review the following parameters, each affecting event collection, or metric data collection, or both. These parameters are preconfigured and it is not necessary to modify the default values. However, if need to, it is possible to modify a value by double-clicking the **Value** column and entering a new value.
<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>max_fetch_interval_min</td>
<td>Maximum time in minutes to go back to for retrieving data.</td>
</tr>
<tr>
<td></td>
<td>Optional value: 180</td>
</tr>
<tr>
<td>Note:</td>
<td>Required for pulling metric data.</td>
</tr>
<tr>
<td>port</td>
<td>The port of the Nagios server.</td>
</tr>
<tr>
<td></td>
<td>Default value is 17778</td>
</tr>
<tr>
<td>Note:</td>
<td>Required for pulling metric data.</td>
</tr>
<tr>
<td>debug</td>
<td>A basic level of logging for the queries and responses between the MID Server and the Nagios system.</td>
</tr>
</tbody>
</table>

7. Click **Test connector** to verify that the MID Server can authenticate itself to the Nagios XI server account to enable communication. If the test fails, follow the instructions on the page to correct the problem and then run another test. The description of the reason for failure is shown in the MID Server log, according to the type of error, such as Events, Metrics, or Bi-Directional.

8. After a successful test, select the **Active** check box and then click **Update**.

9. Optional: **Disable event collection**.

10. The Nagios metrics connector uses the servicechecks and hostchecks tables in the Nagios XI system to pull metrics data. If data is not flowing into the servicechecks and hostchecks tables in the Nagios XI system, complete the following steps. For more information about configuring the Nagios XI server to use individual options instead of data_processing_options, see [ND Москвы - Data Processing Options in ndomod.cfg](#) in the Nagios Support Knowledgebase.

   a) On the Nagios server, edit the file `/usr/local/nagios/etc/ndomod.cfg`.

   b) Comment out the data_processing_options line by adding `#` at the beginning of the line:
   ```
   #data_processing_options=67108669
   ```

   c) Paste the following snippet after the line you just commented out:
   ```
   acknowledgement_data=1
   adaptive_contact_data=1
   adaptive_host_data=1
   adaptive_program_data=1
   adaptive_service_data=1
   aggregated_status_data=1
   comment_data=1
   contact_status_data=1
   downtime_data=1
   event_handler_data=1
   external_command_data=1
   flapping_data=1
   host_check_data=0
   host_status_data=1
   log_data=1
   main_config_data=1
   notification_data=1
   object_config_data=1
   process_data=1
   ```
program_status_data=1
retention_data=1
service_check_data=0
service_status_data=1
statechange_data=1
system_command_data=1
timed_event_data=0

d)  Save the file.
e)  Restart the Nagios service:
    service nagios restart

1. **Create or configure data source registration** to register new metric types incoming from the Nagios system.

2. **Enable or disable metrics** to limit metric collection from the Nagios system to only specific data that is needed.

3. To ensure that metric data is pulled:
   - Verify that **Collect metrics** in the connector definition is selected.
   - Verify that **Metrics collection** in the connector instance is selected.
   - Verify that **Registration mode** in the metric type registration is set to **Active**.

### Create configuration settings rule

Configuration settings affect how metric data is processed. Configuration settings rules override the default values of these configuration settings, letting you build efficient statistical models for different CI classes.

Role required: evt_mgmt_admin

A number of configuration settings determine the behavior of Operational Intelligence MID Servers. In the base system, these configuration settings are configured with default values, data types, and range of valid values. You cannot directly modify these configuration settings or add new ones. However, you can create a metric configuration rule with new configuration settings that override the default values on the MID Servers.

Then, manually apply these rules to all Operational Intelligence MID Servers in a single synchronization operation, or rely on an hourly system job to perform the synchronization.

To take effect, Operational Intelligence MID Servers must be synchronized with the updates to the configuration settings rules.

It is valid to have multiple rules for a setting that affect the same CIs, in which case:

- **Rules in which a filter is defined take precedence over a global rule in which no filter has been defined.**
- **If multiple rules that affect the same set of CIs have the same priority, then only the latest rule to be defined is applied.**
- **If multiple rules with different priorities affect the same set of CIs, then rules with the highest priority are applied.**

1. **Navigate to Operational Intelligence > Configure > Metric Config Rules.**
2. On the Metric Configuration Rules pane, click **New**, and fill out the form.
Metric Configuration Rules form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Rule name.</td>
</tr>
<tr>
<td>Order</td>
<td>Rule priority within all other rules. Higher numbers represent higher priorities.</td>
</tr>
<tr>
<td>Filter by</td>
<td>Check box for displaying the <strong>Rule</strong> field, where you can specify conditions that CIs must meet for the rule to apply. For example, in the <strong>choose field</strong> list, select <strong>Add Related Fields</strong> and then add the filter (<code>class(is)('Linux Server')</code>). If clear, the rule applies globally to all CIs in the Metric To CI Mapping (sa_metric_map) table.</td>
</tr>
</tbody>
</table>

3. Right-click the form title, and click **Save**.

4. In the Metric overridden configurations form section click **New**, fill out the form, and then click **Submit**.

Metric overridden configurations form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Configuration setting for which to override its value. Click the <code>🔍</code> icon to display the list of all configuration settings. Click the <code>ℹ️</code> icon to display the <strong>Metric Settings</strong> dialog with details such as range of possible values. See the following tables (Configuration Settings and Internal Configuration Settings) for details about configuration settings.</td>
</tr>
<tr>
<td>Rule</td>
<td>Rule to which the created configuration setting applies.</td>
</tr>
<tr>
<td>Value</td>
<td>New value that overrides the default value for the specified configuration setting.</td>
</tr>
</tbody>
</table>

You can modify the following configuration settings in the **Name** field.

**Note:** The filter specified in the metric configuration rule does not apply to settings with a global scope.
## Configuration Settings

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>anomaly_detection_enabled</td>
<td>true</td>
<td>boolean</td>
<td>CI/Metric</td>
<td></td>
</tr>
<tr>
<td><strong>Note:</strong> If anomaly_detection_action_level is set, then anomaly_detection_enabled is ignored.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>anomaly_detection_action_level</td>
<td>choices</td>
<td>choice</td>
<td>CI/Metric</td>
<td></td>
</tr>
<tr>
<td>buffer_anomaly_eviction_size</td>
<td>60±1440</td>
<td>integer</td>
<td>Global</td>
<td></td>
</tr>
<tr>
<td>buffer_ci_score_eviction_size</td>
<td>60</td>
<td>integer</td>
<td>Global</td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>------------------------------------------------------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td>buffer_metric_eviction_size</td>
<td>60±1440</td>
<td>60</td>
<td>integer</td>
<td>Global</td>
</tr>
<tr>
<td>Maximum number of metrics that can be stored in internal buffer before sending them to instance for every CI/Metric pair.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>connection_login_timeout_secs</td>
<td>30±60</td>
<td>30</td>
<td>integer</td>
<td>Global</td>
</tr>
<tr>
<td>Maximum time in seconds to log in to the local database on MID Server.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>corrupt_data_count_threshold</td>
<td>10±100</td>
<td>30</td>
<td>integer</td>
<td>Global</td>
</tr>
<tr>
<td>Minimum number of training points (15-minute averages) required to do any statistical analysis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>deprioritize_early_batching_of_anomalous_ci</td>
<td>NA</td>
<td>false</td>
<td>boolean</td>
<td>Global</td>
</tr>
<tr>
<td>Send anomalous CI information immediately or at regular interval.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>max_pool_connections_size</td>
<td>10±50</td>
<td>25</td>
<td>integer</td>
<td>Global</td>
</tr>
<tr>
<td>Maximum number of connections for local database pool.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>observation_time_min</td>
<td>1±1440</td>
<td>1</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Expected minimum metric observation interval.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>robust_central_percentage</strong></td>
<td>50-100</td>
<td>90</td>
<td>double</td>
<td>Global</td>
</tr>
<tr>
<td>Percentage of the residual data to compute the residual standard deviation, used for outlier detection. When set to 100 - uses the regular sample standard deviation.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>sparse_gap_fraction_threshold</strong></td>
<td>0-100</td>
<td>50</td>
<td>double</td>
<td>Global</td>
</tr>
<tr>
<td>If more than this percentage of data is missing and no other class has been identified, classify as SPARSE. Do not attempt to fit a WEEKLY model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>weekly_model_min_days</strong></td>
<td>14-90</td>
<td>15</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Number of days for which data must be available in order to consider only a WEEKLY seasonality decomposition.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>daily_model_min_days</strong></td>
<td>2-90</td>
<td>3</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Number of days for which data must be available in order to consider only a DAILY seasonality decomposition.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>build_snpm_model</strong></td>
<td>NA</td>
<td>true</td>
<td>boolean</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Enable/disable building an SNPM data model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>snpm_minimum_data_count</td>
<td>0.0 e9</td>
<td>5000</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
</tbody>
</table>

Minimum number of data points required for building a stationary nonparametric model.

The following configuration settings are for internal usage.

**Internal Configuration Settings**

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>anomaly_memory_time_min</td>
<td>1±600</td>
<td>45</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
</tbody>
</table>

Anomaly score calculator parameter: Memory time for abnormal situation.

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>excess_z_score</td>
<td>0–3</td>
<td>0.8</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
</tbody>
</table>

Anomaly score calculator parameter: Minimal anomalousness accumulated for outlier.

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear_accumulator_threshold</td>
<td>0.5±5</td>
<td>1</td>
<td>double</td>
<td>Global</td>
</tr>
</tbody>
</table>

Decision Tree Threshold: ACCUMULATOR analysis

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>low_freq_power_threshold</td>
<td>0±100</td>
<td>50</td>
<td>double</td>
<td>Global</td>
</tr>
</tbody>
</table>

Decision Tree Threshold: WEEKLY analysis

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>low_variability_threshold</td>
<td>0.0000000001–0.001</td>
<td>0.0001</td>
<td>double</td>
<td>Global</td>
</tr>
</tbody>
</table>

Decision Tree Threshold: TRENDY analysis

<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>mid_freq_power_threshold</td>
<td>0–300</td>
<td>33</td>
<td>double</td>
<td>Global</td>
</tr>
</tbody>
</table>

Decision Tree Threshold: WEEKLY analysis
<table>
<thead>
<tr>
<th>Name and Description</th>
<th>Values</th>
<th>Default</th>
<th>Data Type</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>multinomial_count_threshold</td>
<td>1</td>
<td>40</td>
<td>integer</td>
<td>Global</td>
</tr>
<tr>
<td>Decision Tree Threshold: MULTINOMIAL analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>non_zero_diff_threshold</td>
<td>0</td>
<td>5</td>
<td>double</td>
<td>Global</td>
</tr>
<tr>
<td>Decision Tree Threshold: NEAR_CONSTANT analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>normal_memory_time_min</td>
<td>0±600</td>
<td>1</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Anomaly score calculator parameter: Memory time for normal situation.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>normal_probability_ewma_timescale_min</td>
<td>0±600</td>
<td>15</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Anomaly score calculator parameter: Normal assessment timescale.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>normal_probability_weight</td>
<td>0±1</td>
<td>1</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Anomaly score calculator parameter: Normal assessment adjustment factor.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sigmoid_offset</td>
<td>0-5</td>
<td>2.1</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Anomaly score calculator parameter: Anomalousness to score conversion.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sigmoid_weight</td>
<td>0-5</td>
<td>1.2</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Anomaly score calculator parameter: Anomalousness to score conversion.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
<td>--------</td>
</tr>
<tr>
<td>tiny_variability_threshold</td>
<td>0±0.001</td>
<td>0.0000000001</td>
<td>double</td>
<td>Global</td>
</tr>
<tr>
<td>Decision Tree Threshold: NEAR_CONSTANT analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>weekly_peak_hi_limit</td>
<td>14±7</td>
<td>10</td>
<td>double</td>
<td>Global</td>
</tr>
<tr>
<td>Decision Tree Threshold: WEEKLY analysis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>weekly_peak_lo_limit</td>
<td>7±0.5</td>
<td>0.7</td>
<td>double</td>
<td>Global</td>
</tr>
<tr>
<td>Decision Tree Threshold: Weekly analysis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>weekly_vs_daily_log_likelihood_threshold</td>
<td>100±600</td>
<td>200</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>By how much log likelihood of weekly needs to be larger than daily, to be the preferred statistical model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>daily_vs_noisy_log_likelihood_threshold</td>
<td>20±600</td>
<td>200</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>By how much log likelihood of daily needs to be larger than noisy, to be the preferred statistical model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>weekly_vs_noisy_log_likelihood_threshold</td>
<td>100±600</td>
<td>200</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>By how much log likelihood of weekly needs to be larger than noisy, to be the preferred statistical model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>trendy_vs_noisy_log_likelihood_threshold</td>
<td>10±600</td>
<td>50</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>By how much log likelihood of trendy needs to be larger than noisy, to be the preferred statistical model.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td>seasonal_loess_width_in_hours</td>
<td>6±24</td>
<td>12</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>seasonal_loess_width_in_hours</td>
<td>Applied to the seasonal component of a weekly or daily model before making a forecast of future behavior. If set to 0, each data point in the seasonal model becomes independent of the rest of the data points.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>robustness</td>
<td>NA</td>
<td>true</td>
<td>boolean</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>robustness</td>
<td>Affects how outliers contribute to seasonal and trend calculations.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>snpm_min_value_threshold</td>
<td>1e9 ± 1e9</td>
<td>0</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>snpm_min_value_threshold</td>
<td>Minimum value of data required for building an SNPM model.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>snpm_max_observation_interval_in_seconds</td>
<td>60 ± 600000</td>
<td>120</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>snpm_max_observation_interval_in_seconds</td>
<td>Maximum expected observation interval required for building an SNPM model.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>min_std_jump_fraction</td>
<td>0.0 ± 1.0</td>
<td>0.2</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>min_std_jump_fraction</td>
<td>Minimum ratio of locally calculated observation noise level to typical jump size that justifies recalculating a larger observation noise variance.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-----------------</td>
<td>---------</td>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td>dynamic_threshold_error_smoothing</td>
<td>dynamic</td>
<td>true</td>
<td>boolean</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Whether to use exponentially weighted moving average to smooth the residuals in the dynamic threshold analysis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ewma_alpha</td>
<td>1e-15 – 1.0</td>
<td>0.02739726027</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>The alpha value of the exponentially weighted moving average in dynamic threshold analysis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic_threshold_beginning_smoothing_length</td>
<td>0.10000</td>
<td>0.050</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Number of smoothed data points to set to the mean of double the smoothing length.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic_threshold_error_buffer_minutes</td>
<td>1 ± 1000</td>
<td>30</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Number of data points around each outlier to group together.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic_threshold_search_start</td>
<td>0.5 ± 20.0</td>
<td>3.0</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Start value at which the optimal control factor is looked for.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic_threshold_search_interval</td>
<td>0.1 ± 5.0</td>
<td>0.5</td>
<td>double</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Interval between search values of optimal control factor.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic_threshold_search_count</td>
<td>1 ± 50</td>
<td>19</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td>Number of values required for searching for optimal control factor.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name and Description</td>
<td>Values</td>
<td>Default</td>
<td>Data Type</td>
<td>Scope</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td><code>dynamic_threshold_error_sequence_limit</code></td>
<td>Maximum number of error groups for a particular control factor value when searching.</td>
<td>1 ± 20</td>
<td>integer</td>
<td>CI/Metric</td>
</tr>
<tr>
<td><code>dynamic_threshold_minimum_data_count</code></td>
<td>Minimum number of raw data points needed before attempting dynamic thresholding.</td>
<td>1 ± 10000</td>
<td>5000</td>
<td>integer</td>
</tr>
<tr>
<td><code>linear_seasonal_log_likelihood_threshold</code></td>
<td>Threshold used in deciding whether to prefer a fitted model with linear seasonality over a model with a periodic component.</td>
<td>10-5000</td>
<td>1000</td>
<td>integer</td>
</tr>
</tbody>
</table>

**Synchronize configuration settings rules**

Operational Intelligence configuration settings rules contain user specified values that override default values that currently exist on Operational Intelligence MID Servers. To take effect, the Operational Intelligence MID Servers must be synchronized with the updated set of configuration settings rules.

Role required: evt_mgmt_admin

You can manually initiate a synchronization to immediately apply the updates on all Operational Intelligence MID Servers. Or, you can rely on a system job, that runs on an hourly recurring schedule, checking for updates to the metric configuration rules. If it is determined that there are updates, the system job sends these updates to the Operational Intelligence MID Servers.

1. Navigate to **Operational Intelligence > Configure > Metric Config Rules**.
2. On the Metric Configuration Rules pane, click **Sync to MID**.
   - Verify that all Operational Intelligence MID Servers have been synchronized with the updated set of configuration settings rules:
     1. Navigate to **Output and Artifacts > ECC Queue**.
     2. In the **Queues** pane, search the **Topics** column for **MetricConfigProbe**.
     3. Check the **Agent** column and verify that all MID Servers are updated.
Configure the action level of anomaly detection

Create a configuration setting rule that refines the level of anomaly detection processing and analysis that is applied to specific CIs and metrics. Set a processing level that reflects the importance of metrics at different stages of implementation, to reduce data load if needed.

Anomaly detection consists of multiple levels of processing. At the lowest level, metric data is gathered from data sources. At the highest level, statistical models are created, anomaly scores are calculated, and anomaly alerts and IT alerts are created. You can apply the full anomaly detection process or a reduced level of processing, for specific data series.

Use the `anomaly_detection_action_level` configuration setting in a configuration setting rule to set the level of processing for specific CIs and metrics. For example, you can initially set only a small percentage of CIs and metrics to generate anomaly alerts, and later increase that percentage.

Action levels

The `anomaly_detection_action_level` configuration setting provides action levels starting at ‘Metrics only’ for basic anomaly processing, and ending at ‘IT Alerts’ for full processing and analysis.

Action levels from lowest to highest:

- **Metrics Only**
  Only gather metrics from the data source without any further processing.

- **Bounds**
  ‘Metrics Only’ processing level and in addition creates statistical model and show bounds in Insights Explorer.

- **Anomaly Scores**
  ‘Bounds’ processing level and in addition calculates and display anomaly scores.

- **Anomaly Alerts**
  ‘Anomaly Scores’ processing level and in addition creates anomaly alerts as applicable.

- **IT Alerts**
  ‘Anomaly Alerts’ processing level and in addition creates IT alerts which are based on anomaly alerts, as applicable. Creating IT alerts which are based on anomaly alerts, is equivalent to having an alert promotion rules.

Specify custom upper and lower metric bounds

Use the Bounds Settings wizard to override learned control bounds by specifying custom upper and lower bounds for an Operational Intelligence metric.

Role required: evt_mgmt_admin

Operational Intelligence calculates upper and lower control bounds for metrics using statistical models learned from historical metric data. Upper and lower bounds are then used in the analysis for detecting anomalous CIs or resources, affecting anomaly scores. Based on historical data, some CIs, resources, or metrics might have values that you specifically want or do not want to be notified about, despite the calculated ranges. For example, trying to meet an SLA without allowing a metric to exceed or be below a certain value for too long without being notified. When concrete upper or lower bound values are known for a metric, you can configure that
metric with custom bounds to override the calculated bounds. For example, a CPU metric which does not exceed 85%.

Use the Bounds Settings wizard to specify custom static bounds or ranges of bounds, and to preview anomaly analysis of actual data based on the custom settings. The metric class and the configuration setting rule that custom bounds must be associated with, are both automatically generated by the Bounds Settings wizard.

In the Bounds Settings wizard, you can choose one of the following options for custom control bounds:

**Static upper and lower bounds**

Custom static upper and lower bound values are used as the control bounds in effect, completely overriding the learned values for upper and lower bounds for the metric. The median of the metric data must be within the specified static bounds range, otherwise the static bounds are not applied.

**Ranges for upper and lower bounds**

Custom upper and lower bounds, each expressed as a range of values. The bounds in effect are a combination of the custom minimum and maximum values that define the bound range, and the learned upper and lower bounds:

- If a learned bound value for a point in time is within the custom range of upper or lower bound, then the learned bound value is in effect.
- If a learned bound value for a point in time is outside the custom range, then the Max or the Min values of the custom range are in effect respectively. If the learned value exceeds the custom range, then the custom Max value is in effect, and if below then the Min custom value is in effect.

When a metric value is outside a control bound, the deviation from the control bound is normalized by a width value that the system calculates from data. Lower and upper width values determine how much a metric value needs to exceed the control bounds to be anomalous. Changing a width for the control bounds affects the speed of reaching a high anomaly score and being notified about the anomaly. In addition to setting custom control bounds, you can override width values by specifying custom deviation and time values that the system uses to calculate custom widths values.

The system stores custom control bounds and custom widths for a metric in a metric class. Custom values in a metric class override the learned bounds and widths for the metric, and are then used in subsequent anomaly analysis.

1. Navigate to **Operational Intelligence > Configure > Bounds Settings**.
2. In the Bounds Settings list view, click **Add** to open the Bounds Settings wizard.
3. On the **Select Scope** tab, configure the following settings and then click **Next**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Metric Type(s)</td>
<td>Source metric type that this bounds setting applies to. Select an item from the drop-down list, and then click the Add Source Metric Type icon (+) to add the selected item to Selected Source Metric Types.</td>
</tr>
</tbody>
</table>
4. On the **Apply Bounds Settings** tab, configure the following settings and then click **Next**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope</td>
<td>Scope of CIs or resources that this bounds setting applies to. You can set the scope to <strong>All CIs</strong>, <strong>Specific CI or class</strong>, or <strong>Specific Resource or Resource class</strong>. Depending on the setting, you can then further filter the scope.</td>
</tr>
<tr>
<td>Bounds setting name</td>
<td>Unique name for this bounds setting.</td>
</tr>
<tr>
<td>Bounds type</td>
<td>Method for specifying the custom static bounds:</td>
</tr>
<tr>
<td></td>
<td>- Set static upper and lower bounds</td>
</tr>
<tr>
<td></td>
<td>- Set ranges for upper and lower bounds</td>
</tr>
<tr>
<td>Show advanced settings</td>
<td>The following fields appear:</td>
</tr>
<tr>
<td></td>
<td>- <strong>If lower bound is</strong> and <strong>And metric value is below lower bound by</strong></td>
</tr>
<tr>
<td></td>
<td>- <strong>If upper bound is</strong> and <strong>And metric value is above upper bound by</strong></td>
</tr>
<tr>
<td></td>
<td>- <strong>Send critical alert after duration (min.)</strong></td>
</tr>
<tr>
<td></td>
<td>Specify the deviation for anomalous values (in absolute values), and the duration for sending an anomaly alert. These values are used internally to calculate the width value for the metric.</td>
</tr>
<tr>
<td>Note:</td>
<td>If the metric class specified for the bounds setting, is configured with width override values from an upgraded release, you cannot set advanced settings. To set advanced settings, accept the notification that appears, agreeing that those width override values be deleted.</td>
</tr>
<tr>
<td>Lower bound</td>
<td>Static lower bound value that replaces the learned lower bound for the metric. Applies to the <strong>Static upper and lower bounds</strong> setting.</td>
</tr>
<tr>
<td>Upper bound</td>
<td>Static upper bound value that overrides the learned upper bound for the metric. Applies to the <strong>Static upper and lower bounds</strong> setting. Value must be greater than <strong>Lower Bound</strong>.</td>
</tr>
<tr>
<td>Lower Bound Range</td>
<td><strong>Min</strong> and <strong>Max</strong> of the custom lower bound range. Applies to the <strong>Set ranges for upper and lower bounds</strong> setting.</td>
</tr>
</tbody>
</table>
ServiceNow    New York    IT Operations Management

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upper Bound Range</td>
<td>Min and Max of the custom upper bound range Applies to the Set ranges for upper and lower bounds setting.</td>
</tr>
</tbody>
</table>

5. On the Preview and Confirm tab:
   a) Select a Source Metric Type and optionally a Metric Type for the preview.
   b) Optionally, enter a CI Name and a Resource Name that you want to include in the preview. If you specify a CI or a resource, it must be within the scope that was previously specified.
   c) Click Preview to perform anomaly detection for the specified metric and CI, using the newly specified custom bounds.
   d) Examine the preview results and if the results are as expected, click Finish. Otherwise, go back to the previous Apply Bounds Settings tab, adjust settings, and then preview again. The preview results are a stripped-down version of anomaly model testing. For more details about the results, and changing chart settings, see Anomaly model testing.

CIs in maintenance mode

Configure anomaly detection to exclude metrics for CIs that are in maintenance mode from model learning.

You can change the settings of system properties to include or exclude metric data for CIs while they are in maintenance mode.

When the system property sa.model_learner.maint_event_record_history_enabled is set to true (default), Operational Intelligence stores historical information about the times CIs enter and exit maintenance mode. When the system property sa_metric.maint_exclusion is set to true (default), metrics from CIs that are in maintenance mode are excluded from model learning.

View metric to CI and resource binding

View the metric to CI and resource binding results, including details for failed bindings which you can use to mitigate the failure. If Operational Intelligence cannot map and bind a metric to a CI or to a resource, then that CI or resource is not included in anomaly detection until it is properly mapped.

Role required: evt_mgmt_admin

If resource binding is not enabled, then no resource binding results appear.

Operational Intelligence uses event rules to map metric data with the specific CI or resource that is associated with the metric event. The results of these mappings are stored in the Metric to CI Mappings (sa_metric_map) table. If mapping fails, then the binding status for the metric indicates the general reason for the failure and further details about the failure are provided. Examine the details for a mapping failure and mitigate the problem to improve accurate and effective metric data processing.

- If mapping to a CI is successful, then the mapped CI appears in the Configuration Item column. If resource binding was attempted but failed, then the mapped CI does not appear, even if the mapping to a CI was successful. Missing or duplicate CIs can cause CI mapping to fail.
If mapping to a resource is successful, then the mapped resource appears in the Resource column. Missing a resource table that is mapped to the CI type to which the event is bound, or a missing resource_path attribute in the event, cause resource binding to fail.

Records in the CI Mappings (sa_metric_map) table remain in effect for 24 hours (by default, if mapping to the CI is not successful), or for 5 days (by default, if mapping to the CI is successful). Later, if within that time period raw data arrives for a metric/CI pair that already has a record, the existing mapping is used to match the data to an existing CI. After Metric to CI Mappings records expire, incoming new raw data requires remapping. These records expire when:

- The 24-hour or 5-day cycle ends.
- An event rule has changed — Triggering an immediate expiration of the respective mapping record.

Adjusting mappings to reflect the addition or removal of CIs by Discovery, takes effect only upon the beginning of the next cycle.

1. Navigate to Operational Intelligence > Configure > Metric to CI.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Data source for the metric.</td>
</tr>
<tr>
<td>Metric Type Id</td>
<td>Metric that is measured for the CI, such as disk% and cpu%.</td>
</tr>
<tr>
<td>Metric Identifier</td>
<td>A string in the form of CI/resource/metric that uniquely identifies a CI and a metric.</td>
</tr>
<tr>
<td>Configuration Item</td>
<td>The CI in the CMDB that is mapped to a metric. A missing value indicates that:</td>
</tr>
<tr>
<td></td>
<td>• The mapping of the metric to a CI was unsuccessful.</td>
</tr>
<tr>
<td></td>
<td>• Resource binding is enabled, but has failed. Configuration Item is missing even though binding to a CI was successful.</td>
</tr>
<tr>
<td>Resource</td>
<td>The resource that is mapped to a metric. If resource binding is enabled, then a missing value indicates that the mapping of the metric to a resource was unsuccessful.</td>
</tr>
<tr>
<td>Expiration Date</td>
<td>Record expiration date.</td>
</tr>
<tr>
<td>Binding Status</td>
<td>Status of the mapping of the metric to a CI or resource.</td>
</tr>
<tr>
<td></td>
<td>A CI Found status indicates that the mapping was successful, in which case the mapped CI appears in the Configuration Item column. Any other status indicates that the mapping failed, and a general reason for the failure is provided. Further details about the failure are provided in the Binding Failure Reason column.</td>
</tr>
<tr>
<td>Binding Failure Reason</td>
<td>Specific details about the mapping that failed, such as class and host names, and the queries that were used. Click the respective Event link to see further details in the Processing Notes field.</td>
</tr>
</tbody>
</table>
2. Examine the **Binding Status** and **Binding Failure Reason** as appropriate. Use the details provided to remediate the problem:

<table>
<thead>
<tr>
<th>Binding Status</th>
<th>Remediation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host CI Missing</td>
<td>Run Discovery or use any other method to populate the CMDB with the missing host CI.</td>
</tr>
<tr>
<td>Non-Host CI Missing</td>
<td>Run Discovery or use any other method to populate the CMDB with the missing non-host CI.</td>
</tr>
<tr>
<td>Non-Host CI Duplicate</td>
<td>Use the <a href="#">Duplicate CI Remediator</a> to remediate the duplication.</td>
</tr>
<tr>
<td>Dependent CI Duplicate</td>
<td>Use the <a href="#">Duplicate CI Remediator</a> to remediate the duplication.</td>
</tr>
<tr>
<td>Dependent CI Missing</td>
<td>Run Discovery or use any other method to populate the CMDB with the missing dependent CI.</td>
</tr>
<tr>
<td>Duplicate Metric Map Entry Exists</td>
<td>Check <strong>Binding Failure Reason</strong> and compare the metric identifiers associated with the duplicate metric map entries. If metric identifiers have changed (for example, an identifier was added), set the entry for this series in the CI Mappings (sa_metric_map) table to expire or delete the entry. Also check the associated event rule and ensure that it extracts the attributes correctly and that it does not extract the same CI for two different metric identifiers. This error might appear when 2 different metric identifiers map to the same CI/metric.</td>
</tr>
<tr>
<td>Resource ID Not Found</td>
<td>Verify that the CI Type To Resource Class (sa_ci_type_to_resource_class) table contains a mapping record for the respective CI type, or for one of its parents. Then ensure that a table with the name of resource class for this record, exists.</td>
</tr>
</tbody>
</table>

Group **Binding Failure Reason** by **Binding Status** to create a summary report with the number of records per failure category. Use this report to efficiently remediate the problems.

**Anomaly model testing**

Use anomaly model testing to apply and evaluate anomaly detection for a small set of CIs and metrics, using actual metric data. Compare test results to expected results, then fine-tune the anomaly detection model before enabling anomaly detection for the tested CIs and metrics in the production environment.

For example, before enabling anomaly detection for Linux Servers, you can use anomaly model testing with actual metric data collected for those Linux Servers. Existing alerts might indicate an anomaly for a specific metric and CI. You can create an anomaly test rule to test anomaly detection for those CIs and metrics, and then check if the expected anomaly is detected. You can also examine the bounds calculated by the anomaly model to see if they match expected bounds. After you fine-tune the test model, you can enable anomaly detection for Linux Servers.
Anomaly model testing supports up to 20 time series, which can include 20 metric types for the same CI, the same metric type for 20 different CIs, or any combination of those options.

**View test results in the Insights Explorer**

Upon completion of a test run, a URL to an Insights Explorer becomes available. The anomaly model testing results appear in this trimmed-down version of the Insights Explorer. This Insights Explorer is pre-loaded with the charts and scores for the metric series specified in the anomaly test rule. You cannot add any CMDB groups or application services to this Insights Explorer.

Each chart in the Insights Explorer, contains the Anomaly Test Statistics section in its **Chart Settings**. This section is specific to anomaly model testing and provides the following options:

- **Show Bounds**: Displays a computed upper and lower bound for the metric in an anomaly model testing, based on learning of past metric values. The upper and lower bounds always appear together.
- **Show Anomaly Scores**: Displays anomaly scores on a virtual axis of 0-10, for an anomaly model test. Color code is based on the score thresholds that are defined in the Anomaly Score to Event Severity Map (sa_metric_anomaly_score_to_event_severity_map) table.

**Use test results to fine-tune anomaly detection**

Anomaly model testing uses the **upper and lower bounds** configured for selected series in the anomaly test rule. The anomaly model test results show how many anomaly alerts would have been generated based on the specified settings, if anomaly detection was enabled for the tested CIs and metrics. You can then decide if this result is acceptable. Fine-tune the bounds so that the number of anomaly alerts that are generated is not excessive, but sufficient to accurately indicate an out of bounds score.

**Create an anomaly test rule**

To use the anomaly model testing, create an anomaly test rule in which you specify up to 20 metric series to test anomaly detection for. Run the anomaly test, and after it completes, use the provided URL to open the Insights Explorer which is pre-loaded with the model testing results.

Role required: evt_mgmt_user

1. Navigate to **Operational Intelligence > View Metric & Anomalies > Anomaly Model Testing**.
2. Select an existing anomaly test rule to edit and run again, or click **New** and fill out the Anomaly Test Rules form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the model test rule.</td>
</tr>
<tr>
<td>Applies to</td>
<td>The table from which to select CI/metric pairs for the model test rule. The table used is Metric To CI Mapping (sa_metic_map).</td>
</tr>
<tr>
<td>Rule</td>
<td>Filter conditions to specify pairs of CIs/metrics for the anomaly model testing. If the rule yields more than 20 pairs of CIs/metrics, then the list is truncated to 20.</td>
</tr>
<tr>
<td>Start date/End data</td>
<td>The time period for which to apply the anomaly detection testing. The time period must be within the past 7 days.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Status of the test, automatically updated by the system during a test run. Status is initially Scheduled and eventually changes to Complete.</td>
</tr>
<tr>
<td>URL</td>
<td>URL to the Insights Explorer in which the anomaly model test results are pre-loaded.</td>
</tr>
<tr>
<td>Notes</td>
<td>Logging summary, automatically updated by the system with details of each test run of the anomaly test rule.</td>
</tr>
</tbody>
</table>

3. **Click Run Anomaly Test.**

   During the test run, the **Status**, **URL**, and **Notes** fields are automatically updated to reflect the progression of the job.

4. **After Status changes to Complete, click the URL link to open the Insights Explorer to see the results of the test.**

The bounds and the chart for the anomaly model test, appear in faint colors in Insights Explorer. Also, the upper and lower bounds that are calculated by the test, use a line pattern that is different than line pattern used for actual anomaly detection results. These differences let you distinguish between actual anomaly detection and anomaly model test results, if both are selected in the chart settings.

- Modify the date range for the test or choose different time series, and click **Run Anomaly Test** again to continue and fine-tune the model.
- Click the **Chart Settings** icon for a chart to toggle the display of statistics and aggregations on the chart. Select **Show Bounds** and **Show Anomaly Scores** in the Anomaly Test Statistics and in the Anomaly Statistics sections, to display test results next to actual anomaly detection results.

**Note:** Anomaly Model Testing results might be very similar to actual anomaly detection results. If you configure the chart to display both at the same time, as you move the mouse over the chart, it might be difficult to identify points of differences.
When you are satisfied with the anomaly test results, enable anomaly detection in the production environment, for the metrics and CIs specified in the test. See Choose and configure metrics to monitor for more details.

View metric values in the Insights Explorer

Operational Intelligence calculates statistics for CI metric data. Insights Explorer displays these metric values as metric charts for the CIs in the CMDB. Insights Explorer lets you overlap any metrics for any CIs in a single chart to create a multi-layered view of metric values across a time range.

Role required: evt_mgmt_user

The Insights Explorer provides a canvas to which you can drag various metrics for CIs and create charts. You can place a single metric on the canvas, for an individual metric for a CI. For a more comprehensive chart, you can layer several metrics for a single CI or for different CIs to create a multi-metric chart. In such layered chart, each metric is distinguished by a unique color, which lets you compare metric values across CIs within a specific time range.

You can also access the Insights Explorer from the following maps:

- View metrics in a Dependency Views map
- View metrics from a business service map
Insights Explorer

If the host name of the CI is available, it appears underneath the CI name. Host name is populated by retrieving the node information from the generated binding event as part of metric binding.

To navigate to a CI for which to display metrics:

- Use Hottest Configuration Items ( ) to quickly access the 10 most anomalous CIs.
Use Configuration Items ( ) to create a separate custom list of any CIs from the CMDB and then add the metrics for these CIs to the canvas.

Use Application Services ( ) to create a custom list of application services that let you drill into the CIs of these services. You can then add metrics for these CIs to the canvas.

Use CMDB Groups ( ) to create a custom list of CMDB Groups that let you drill into the CIs of these groups. You can then add metrics for these CIs to the canvas.

The following colors, which are based on color settings for alert severities, indicate anomaly severities:

- **Red**
  Critical severity (highest severity).

- **Orange**
  Major severity.

- **Yellow**
  Minor severity.

- **Blue**
  Warning severity (lowest severity).

- **Green**
  Informational. No severity.

The dots at the upper left on the CI icons indicate general anomaly tracking status:

- Grey: Metrics are being tracked for the CI, but none of those metrics have ever had an anomaly.
- Green: Metric for CI had at least one anomaly in the past, but anomaly score is currently 0.
- Colored: Corresponds to the last recorded severity level according to the configuration of the color band ranges.
- No dot: There are no metrics for the CI and there is no data to drill into.

*Note: Configuration Items and Application Services are likely to include CIs for which there is no metric data.*

Operational Intelligence does not track changes to an anomaly score if it is within the green band range.

As you create charts, a legend for each series chart automatically appears underneath the chart. For a single-metric chart, the legend has entries for the aggregations in the chart. For a multi-metric chart, the legend has entries for the metrics in the chart.

Modifications in the Insights Explorer remain during the session, and once the Insights Explorer is refreshed it returns to its initial state.

1. Navigate to **Operational Intelligence > View Metric & Anomalies > Insights Explorer**.
2. Click a CI to drill down to its associated list of metrics, ordered by their most recent anomaly scores.
For each metric, a sparkline displays metric values for the CI for the last hour. Point to the sparkline to display the metric value at each point of time.

If resource binding is enabled, then the list of series being monitored for a CI also contains series that belong to a resource and metric. The naming convention for such series is `<resource name>/<metric name>`. For a series that is only bound to a CI, the naming convention is `<metric name>`.

3. Drag a metric to the Drop Metric To Create Chart area on the canvas. Point to the chart to display the exact metric value at each point of time. The metric chart displays data for the specified time period, according to the chart settings. By default, raw data is displayed. If the time period is longer than a week or earlier than the last week, then average data is displayed.

Drag subsequent metric to the canvas as follows:

- In the Drop Metric To Create Chart area, so it is displayed individually in a new chart.
- Onto an existing chart, in the Replace Chart highlighted block to overwrite the existing chart.
- Onto an existing chart, in the Add Metric highlighted block to overlap it with the existing metrics in the chart.

When you display multiple metrics in a single chart, the same aggregation selection applies to all the metrics, and statistics settings are not available. For example, if Average is selected in the Chart Settings, then averaged values are displayed for all the metrics in the chart.

4. Use the Search box to search and filter for specific configuration items, application services, or CMDB groups that are not listed by default. In the SEARCH RESULTS section, click the + icon next to an item that you want to add to the ADDED ITEMS section. Or, use advanced search:

- Select and then click Add Configuration Item to add any specific CIs. In the Add Configuration Items dialog box, select a CI Class Type such as Linux Server. Then specify a filter condition such as (Operating System) is (Linux Red Hat) to narrow down the search.

- Select and then click Add Application Service to add all CIs included in a specified application service. In the Add Application Services dialog box, select Discovered Service or Manual Service as the CI Class Type and then specify filter conditions to narrow down the search.

  Note: You can add an application service only if its Operational Status is "operational".

- Select and then click Add CMDB Group to add all CIs included in a specified CMDB group. In the CMDB Groups dialog box specify filter conditions such as (GroupName) starts with (d), to narrow down the search.

After selecting the items that you want to add, click Add.

5. Add related CIs:

a) Click the Add related CIs icon next to a CI.

The Add Related Configuration Items dialog box appears, displaying all relationships in which the selected CI is a parent or a child.
b) To further filter the list, you can define specific conditions that must be met.

c) Select the relationships from which you want to add the parent or the child CI, whichever is not the selected CI.

d) Click **Add**.

The CIs from the selected relationships, which are not the selected CI, are added to the list of CIs in the Insights Explorer.

6. Point to a legend item and click its 'X' icon to remove the corresponding metric from the chart, or to disable the corresponding series from the Chart Settings.

7. Click the cog **Settings** icon to open application settings:
   - Enable, disable, or set the frequency for refreshing anomaly data on the Insights Explorer.
   - Activate or deactivate the display of related lists such as All Alerts, Incidents, and Problems. For more information, see [Create or modify map indicators](#).

8. Click the **Chart Settings** icon to toggle the display of statistics and aggregations on the chart. Enabling or disabling an item to add or to remove metrics from a chart, also updates the legend of the chart to reflect the change.

<table>
<thead>
<tr>
<th>Chart Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show Bounds</td>
<td>Displays a computed upper and lower bound for the metric, based on learning of past values. The upper and lower bounds always display together.</td>
</tr>
<tr>
<td>Show Anomaly Scores</td>
<td>Displays anomaly scores on a virtual axis of 0-10. Color code is based on the score thresholds that are defined in the Anomaly Score to Event Severity Map (sa_metric_anomaly_score_to_event_severity_map) table.</td>
</tr>
<tr>
<td>Average</td>
<td>Displays aggregated average metric values calculated in one hour intervals. Aggregated averages are displayed as follows:</td>
</tr>
<tr>
<td></td>
<td>• For data that is up to 8 days old: Displays 10-minute aggregation windows.</td>
</tr>
<tr>
<td></td>
<td>• For data that is up to 94 days old: Displays 10-minutes aggregation windows.</td>
</tr>
<tr>
<td></td>
<td>• For older data: Displays 1-hour aggregation windows.</td>
</tr>
<tr>
<td>Minimum</td>
<td>Displays aggregated minimum metric values calculated per one hour time periods. Aggregated minimum values are displayed as follows:</td>
</tr>
<tr>
<td></td>
<td>• For data that is up to 8 days old: Displays 10-minute aggregation windows.</td>
</tr>
</tbody>
</table>
### Chart Setting

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
</table>
| Maximum | Displays aggregated maximum metric values calculated for one hour time periods. Aggregated maximum values are displayed as follows:  
  - For data that is up to 8 days old: Displays 10-minute aggregation windows. |
| Raw | Displays the raw, unaggregated metric values. These values are kept only for the last 8 days. |

- Point to the icon of a CI in the right-hand side bar to display the CI type.
- Zoom in or out by changing the default **Last Hour** time range selection. Select one of the preset time periods to display anomaly scores for the last **6 Hours** for example, or specify a custom time period for up to 90 days back.

To zoom in, highlight a section of a chart by pointing to the upper left corner of the section and dragging the mouse device to the lower right corner of the section. The time range of the chart changes accordingly. If there are other charts on the canvas, they are all automatically synchronized to display data for the same time range. The time range in the title bar automatically updates to reflect the new time range.

- Right-click on a CI in the right-hand side bar and select **View Form** to open its CI form.
- Click the map icon next to an application service listed in the **Application Services** tab, to open its application service map.
- Click Remove (_remove_) next to a CI to remove it from the Insights Explorer.
- Click the Export icon in a chart to download it as a .png or .svg image, or as a .pdf document.

### Insights Explorer Settings

You can configure Insights Explorer settings to control for example, the refresh frequency or the display of related records.

Click the cog **Settings** icon to open the Application settings dialog box.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh interval</td>
<td>Enable and set the frequency for refreshing metrics data on the Insights Explorer.</td>
</tr>
</tbody>
</table>
| Indicators  | Enable any indicator to display related records such as incidents or problems for CIs. The respective icon displays on the timeline in the metric chart, indicating the number of records for the CI.  
  A toggle is available only if the **Active in Metrics** field is selected on the Map Indicator form for the respective table. For example, the **Incident** toggle is available only if **Active in Metrics** is selected on the Map Indicator form for the incident (Incident) table. |
Create an Insights Explorer view

You can create a custom view in the Insights Explorer that saves any metric charts that were added to the canvas, and any added configuration items. After you create a view, you can share a link to the view.

1. Navigate to **Operational Intelligence > View Metric & Anomalies > Insights Explorer.**
2. Click **Create View** ( ), and enter a **Name** in the Create New View dialog box. If you are already creating a view, then you can choose **Create With Existing** to load an existing view to work with rather than starting with a blank canvas.
3. Click **Create**.
4. Add items to be included in the view:
   a) Drag metrics and create charts on the canvas.
   b) Add configuration items.

While modifying a view, it is automatically saved every time you update it. The time settings are not saved in views, and when a saved view is loaded, its time period is always set to **Last hour** displaying current information.

- Share a view: Click **Share View** ( ), and then in the Share dialog box, click **Copy Link**. Send or paste the link to make it accessible to whoever you want to share the view with. A shared link captures the time range of the view, the charts that were added to the canvas, and any added configuration items.
- Display a saved view: Open the **Select a Saved View** drop down list, and select a previously saved view that you want to display.

**Anomaly alerts**

Anomaly alerts indicate deviation from projected metric values for monitored CIs. Anomaly alerts are separate from the regular IT alerts, and are not displayed in the Alert Console. You can define an anomaly alert promotion rule to generate an IT alert that is based on anomaly alerts.

Role required: **evt_mgmt_user**

The statistical model is used to calculate standard deviations, upper and lower bounds, and statistical outliers which are then used to detect anomalies. An anomaly is when metric values are out of the projected values according to the statistical model. The system monitors the frequency and persistence of statistical outliers across time to compute a score between 0-10 that indicates how abnormal a deviation is.

Operational Intelligence constantly generates anomaly alerts whenever the anomaly score is above zero. If there is a score that is above four and which has changed from the previous score, then it is sent to the instance. On the instance, the entire sequence of anomaly scores over time can then be displayed in the Insights Explorer.

You can view a list of anomaly alerts as described below. For summaries and reports, you can view the **Anomaly Map** and the **Anomaly Alert Dashboard**.

1. Navigate to **Operational Intelligence > View Metric & Anomalies > Anomaly Alerts.**
2. In the **Alerts Anomalies** list view, double-click an alert that you want to view.

Information about an anomaly alert includes details such as the metric value, promotion parameter, and upper and lower bounds. Description includes details about why the
anomaly alert was generated, and other details such as the metric name and the anomaly score.

- You can [create an alert promotion rule](#) to generate regular IT alerts that are based on anomaly alerts.
- Right-click an alert, and then click **View Metrics** to open the integrated Insights Explorer and Dependency Views map for the CI associated with the alert.

**View Anomaly Map**

Anomaly Map displays the 10 CIs with the highest metric anomaly scores across a time period. The map lets you drill down into each CI to display anomaly scores for each of the CI’s metrics. Anomaly scores are color coded to help you isolate problem areas at a glance.

Role required: evt_mgmt_user

Operational Intelligence monitors metric data for CIs and detects anomalies when values are out of bounds, according to the statistical model. Anomalies are scored on a scale of 0-10 and mapped into color codes based on score thresholds defined in the Anomaly Score to Event Severity Map (sa_metric_anomaly_score_to_event_severity_map) table.
If the CIs host name is available, it appears underneath the CI name. Host name is populated by retrieving the node information from the generated binding event as part of metric binding.
To navigate to the CI for which to display metrics:

- Use Hottest Configuration Items ( ) to quickly access the 10 most anomalous CIs.
- Use Configuration Items ( ) to create a separate custom list of any CIs from the CMDB that have metric data.
- Use Application Services ( ) to display all the CIs included in a specified application service.

**Note:** Some CIs in an application service might not have metric data. In this case, the respective icon does not have a dot on the upper left side, and no metric data is displayed.

- Use CMDB Groups ( ) to display all the CIs included in a specified CMDB group.

1. Navigate to **Operational Intelligence > View Metric & Anomalies > Anomaly Map**. In the Anomaly Map, point to a colored anomaly box to display the details of the CI and the time period for the anomaly scores.
   The color of each anomaly box represents the highest anomaly score of all metrics for the CI for the time period.
2. Click a CI in the left-hand side bar to display a breakdown of its metrics with the respective anomaly scores across the time period.
   If resource binding is enabled, then the list of series being monitored for a CI also contains series that belong to a resource and metric. The naming convention for such series is `<resource name>/<metric name>`. For a series that is only bound to a CI, the naming convention is `<metric name>`.
3. Point to an anomaly box to display its precise anomaly score.
   The color coded anomaly score next to the metric represents the latest anomaly score for the CI.
4. Use the Search box to search and filter for specific configuration items, application services, or CMDB groups that are not listed by default. In the SEARCH RESULTS section, click the + icon next to an item that you want to add to the ADDED ITEMS section. Or, use advanced search:
   - Select and then click **Add Configuration Item** to add any specific CIs. In the Add Configuration Items dialog box, select a CI Class Type such as Linux Server. Then specify a filter condition such as (Operating System) (is) (Linux Red Hat) to narrow down the search.
   - Select and then click **Add Application Service** to add all CIs included in a specified application service. In the Add Application Services dialog box, select Discovered Service Or Manual Service as the CI Class Type and then specify filter conditions to narrow down the search.

**Note:** You can add an application service only if its Operational Status is 'operational'.
Select and then click Add CMDB Group to add all CIs included in a specified CMDB group. In the CMDB Groups dialog box specify filter conditions such as (GroupName) (starts with) (d), to narrow down the search.

After selecting the items that you want to add, click Add.

- Point to a CI's icon in the left-hand side bar to display the CI type.
- Click for a CI to drill to the CI's metrics. Click next to a metric to view the metric in the Advanced Insights Explorer.
- Click the cog Settings icon to open application settings and to configure the Refresh interval. Enable and set the frequency for refreshing anomaly data on the Anomaly Map.
- Zoom in or out by changing the default Last Hour time range selection. Select one of the preset time periods to display anomaly scores for the last 6 Hours for example, or specify a custom time period for up to 90 days back. Click the timestamp at a column heading to drill into the time period between the current column and the column to the right.
- Right-click a CI in the left-hand side bar and select View Form to open its CI form.
- Click a tile in the Anomaly Map to switch to the Insights Explorer, which displays the metric chart for the CI that was clicked.

Anomaly Alert Dashboard

View important anomaly alert reports in a single dashboard. Anomaly Alert Dashboard provides summaries about anomaly alerts and promoted alerts, in relation to Event Management alerts.

The reports in the Anomaly Alert Dashboard use Performance Analytics indicators and Reporting, and require that Performance Analytics (com.snc.pa) is activated.

To access the Anomaly Alert Dashboard, navigate to Operational Intelligence > Reports > Anomaly Alert Dashboard.

Role required to view reports: evt_mgmt_user

% Daily Alerts are Promoted Anomalies

Shows the percentage of newly promoted alerts, in relation to the overall new IT alerts, for the time period. The change in minutes from the previous report is also available. Click the tile to drill down for more details.

Avg Resolve Time of Promoted Alerts

Shows the average number of hours it took to resolve anomaly alerts that were promoted to IT alerts within the time range. The percentage point change from the previous report is also available. Click the tile to drill down for more details.

Number of Anomaly Alerts

Shows the number of anomaly alerts that were created per day in the time range, broken down by promotion to Event Management alerts.
Bars in the chart are broken down into sections which are color coded by the associated alert type:

- Anomaly alerts that were promoted to Event Management alerts.
- Anomaly alerts that were not promoted to Event Management alerts.

To view further details:

- Click Promoted or Not Promoted in the report legend, to show or hide details for an alert type.
- Point to a section on a bar to show the date and the number of promoted or not promoted anomaly alerts for the specified day.
- Click a bar to view a detailed scorecard for the bar.

Create anomaly alert promotion rule

Anomaly alerts are generated by Operational Intelligence to indicate deviation from projected metric values for monitored CIs. You can create an anomaly alert promotion rule to generate a regular IT alert that is based on an anomaly alert.

Role required: evt_mgmt_admin

Anomaly alerts are separate from regular IT alerts and they do not appear in the Alert Console. However, anomaly alerts that were promoted to IT alerts do appear in the Alert Console.

You can create an [anomaly alert promotion rule blacklist](#) to exclude the promotion of alerts for specific CIs that would otherwise be promoted by an anomaly alert promotion rule.

1. Navigate to Operational Intelligence > View Metric & Anomalies > Anomaly Alerts.
2. Right-click an alert in the Alert Anomalies list and select Promote Anomaly Alert.
3. Fill in the fields on the Alert promotion rule form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the anomaly alert promotion rule.</td>
</tr>
<tr>
<td>Promotion type</td>
<td>A filter that defines the scope of anomaly alerts for which the rule applies to. Filtering anomaly alerts can be based on:</td>
</tr>
<tr>
<td></td>
<td>• Metric name: The value of Source_metric_type in the Additional_info field in the anomaly alert.</td>
</tr>
<tr>
<td></td>
<td>• CI type: Type of the configuration item in the alert.</td>
</tr>
<tr>
<td></td>
<td>• Configuration item: Configuration item in the alert.</td>
</tr>
<tr>
<td></td>
<td>• Promotion parameter: In Additional Info of the anomaly alert, use the field specified in Field name for anomaly promotion in the Metric Registration table.</td>
</tr>
<tr>
<td>Source</td>
<td>Data source that is monitoring the metric type.</td>
</tr>
<tr>
<td>Metric name</td>
<td>The raw metric to filter anomaly alerts by. This field appears only when Promotion type is set to Metric name.</td>
</tr>
<tr>
<td>CI type</td>
<td>The CI type to filter anomaly alerts by. This field appears only when Promotion type is set to CI type.</td>
</tr>
<tr>
<td>Configuration item</td>
<td>The configuration item to filter anomaly alerts by. This field appears only when Promotion type is set to Configuration item.</td>
</tr>
<tr>
<td>Regular expression for the promotion parameter field</td>
<td>The regex applicable for the specified source, to filter anomaly alerts by. This field appears only when Promotion type is set to Promotion parameter.</td>
</tr>
<tr>
<td>Alert</td>
<td>The anomaly alert which was used to initiate the creation of this rule.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Active</td>
<td>Select to enable the rule.</td>
</tr>
<tr>
<td>Minimal score</td>
<td>A threshold value between 0-10 that the anomaly score of an anomaly alert</td>
</tr>
<tr>
<td></td>
<td>must be equal to or greater than, for the anomaly alert to be promoted.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Additional information</strong> field in an anomaly alert form contains the</td>
</tr>
<tr>
<td></td>
<td>alert's anomaly_score value.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

After the anomaly alert promotion rule runs for an event that matches the rule criteria, a new Event Management alert is created that appears in the Event Management dashboard. Also, the anomaly alert that the new Event Management alert is based on, is updated with details about the promotion. To examine these details:

1. Navigate to **Operational Intelligence > View Metric & Anomalies > Anomaly Alerts**.
2. Double-click an alert that has been promoted.
   - Click **Processing Notes** to display processing details.
   - Click **History** to display details about the alert promotion.

Add a blacklist item to prevent the creation of a new Event Management alert that would otherwise be created by an anomaly alert promotion rule. Right-click an anomaly alert and select **Blacklist CI**.

**Create anomaly alert promotion rule blacklist**

You can create a blacklist of CIs to be excluded from promotion to regular alerts. Alerts related to the CIs in the blacklist would otherwise be promoted to regular alerts by an anomaly alert promotion rule.

*Anomaly alert promotion rules* must exist, otherwise, blacklist CIs have no effect.

Role required: evt_mgmt_admin

Anomaly alert promotion rules promote anomaly alerts to regular alerts. Alerts generated by CIs that are included in the blacklist, are not promoted even if they satisfy the anomaly alert promotion rule filter.

1. Navigate to **Operational Intelligence**.
2. Open the **Alert Promotion Rules Blacklist** by doing either of the following:
   - Navigate to **Settings > Anomaly Alert Promotion CIs Blacklist**, and then click **New** or open an existing rule.
   - Navigate to **View Metric & Anomalies > Anomaly Alerts**, right-click an alert in the **Alert Anomalies** list, and then select **Blacklist CI**.
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration item</td>
<td>The CI from the alert, that is to be added to the CIs blacklist.</td>
</tr>
<tr>
<td>Source</td>
<td>Data source of the alert.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.
Use Advanced Insights Explorer to query CI metric data

Advanced Insights Explorer lets you query metric data for CIs for which metric data is collected. CIs for the query can be from different sources such as application services, relationships, or a list of anomalous CIs. Query results are displayed in charts for further understanding of system behavior.

Role required: evt_mgmt_user

Use Advanced Insights Explorer to create queries such as:

- Server CPU metric value that is higher than a given value.
- Metrics for all Apache Servers in which CPU utilization is greater than 90% and anomaly score is greater than 8 for a given time period.
- Metrics for the top 10 Apache Servers with CPU utilization of 80%.

A query in the Advanced Insights Explorer consists of a query title, a time range, a combination of one or more guided expressions or script expressions, and the resulting chart of the query. Each guided expression or script expression defines a set of CIs and a metric type whose values are retrieved for the set of CIs. The Advanced Insights Explorer utilizes the platform scripting capability to run the queries.

**Guided Expression**

A query expression that you create by using a wizard. After you complete the wizard, the query is converted into a JavaScript code snippet that can run using the platform scripting capability.

**Script Expression**

A query expression that is based on a direct entry of JavaScript. You can create a custom script, which, for example, queries for data that is not supported by default. You can create a custom function to use in a script expression.

Some of the data that the Advanced Insights Explorer queries, is stored as time series. The Advanced Insights Explorer uses metric data time series, anomaly scores time series, and upper and lower bounds time series. The result of a query then is a single time series or a set of time series that satisfy the criteria specified in the query expressions.

Open the Advanced Insights Explorer, by navigating to Operational Intelligence > View Metric & Anomalies > Advanced Insights Explorer. Then do any of the following steps.

- In the Queries sidebar, select a query to edit or click New Query and then select the new query to edit.
- Add a guided expression to a query:
  a) If the Guided Expression tile is not visible, click Add Expression.
  b) Click Create Expression on the Guided Expression tile to open the Create An Expression dialog box.
  c) Configure the settings on the Select Data Source tab to define the data set and to select the metric type for the query, and then click Next.

Your selection of CI Source determines which other fields appear.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Source</td>
<td>Source from which to select CIs for the query. Most options are tables such as Metric table and Service table. Another option is Existing Guided Expression.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CIs</td>
<td>Filters the CIs from a table selected in CI Source to a specific class. You can add other conditions that CIs must satisfy.</td>
</tr>
<tr>
<td>Resource</td>
<td>Toggle to query for series belonging to resources for a CI class. Click the Resource filter icon to filter on resources.</td>
</tr>
<tr>
<td>Parent or Child</td>
<td>Filters the set of CIs for the query to those CIs that are parent or child CIs in the relationships in the CI Relationship (cmdb_rel_ci) table.</td>
</tr>
<tr>
<td></td>
<td>Applies when CI Source is set to Relationships table.</td>
</tr>
<tr>
<td>Guided Expression</td>
<td>Limits the CIs for the query to the results of an existing guided expression.</td>
</tr>
<tr>
<td></td>
<td>When CI Source is set to Relationships table, the specified guided expression further filters the set of CIs by joining the resulting CIs from the guided expression to the opposite of what is selected in Parent or Child.</td>
</tr>
<tr>
<td></td>
<td>For example: Parent or Child is set to Parent and a guided expression is specified. The list of records from which the parent CI is selected in the cmdb_rel_ci table, is further limited to only those records whose child CI is included in the results of the specified guided expression.</td>
</tr>
<tr>
<td></td>
<td>Also applies when CI Source is set to Relationships table.</td>
</tr>
<tr>
<td>Anomaly Score</td>
<td>Filters the set of CIs for the query to those CIs whose anomaly score satisfies the Anomaly Score criteria. Anomaly score values range from 0—10.</td>
</tr>
<tr>
<td></td>
<td>Applies when CI Source is set to Anomaly score table.</td>
</tr>
<tr>
<td>Metric Type</td>
<td>Metric type for which to retrieve data for the set of CIs in the query.</td>
</tr>
</tbody>
</table>

d) Configure the settings on the Filter Data Series tab to filter and sort anomaly score and data time series for the query. Then click Next.

You can independently set the Data Filter and the Anomaly Score Filter. Both filters can be enabled or disabled, or one of them can be enabled and the other disabled.

Each time series from the Select Data Source tab is filtered through the data filter and the anomaly score filter, if enabled. Only time series that satisfy the enabled filters, are included in the data set for the query. If both filters are disabled, then the entire data set from the Select Data Source tab, is included in the query without filtering.
### Data Filter/Anomaly Score Filter

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregation</td>
<td>Aggregation type (such as average, minimum, or maximum) to use for aggregating each time series.</td>
</tr>
<tr>
<td>Start date</td>
<td>Time range of the time series to which the specified Aggregation is applied.</td>
</tr>
<tr>
<td>End date</td>
<td></td>
</tr>
<tr>
<td>Condition</td>
<td>Condition that the aggregated time series needs to satisfy to be included in the query.</td>
</tr>
<tr>
<td>Value</td>
<td></td>
</tr>
</tbody>
</table>

### Limit and Sort

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit Data Series</td>
<td>The number of data series to use for the query, counting from the top of the set of filtered and sorted data series.</td>
</tr>
<tr>
<td>Sort</td>
<td>The sort order for the resulting CIs. If both, the data filter and the anomaly score filter are enabled, then choose which of those filters to use for sorting the results.</td>
</tr>
</tbody>
</table>

### Transform Options

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group by</td>
<td>A CI column (attribute) by which to group the query results. For example, the datacenter attribute of the CI.</td>
</tr>
<tr>
<td>Transform Output</td>
<td>Aggregation type to apply when aggregating all the time series in each group (as specified in Group by) into a single time series. Click the '+' or '-' to add or remove transforms to apply to the query results.</td>
</tr>
</tbody>
</table>

### Output Types

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric</td>
<td>Include charts of metric time series in the query results.</td>
</tr>
<tr>
<td>Bounds</td>
<td>Include charts of upper and lower bounds time series in the query results.</td>
</tr>
</tbody>
</table>
Add a script expression to a query:
   a) If the Script tile is not visible, click Add Expression.
   b) Click Create Expression in the Script tile.
   c) Type in a custom Java script.

Query actions:
   - Click Run to run a query.
   - Click Save to save a query.
   - Point to specific areas on the report graph to show details specific to that time.
   - Use Search in the Queries sidebar, to search for a query.
   - Click a query title in the header of the query pane to rename a query.
   - Set a time range for a query by overwriting the default start and end times in the header of the query pane.
   - Click to export the report in a PNG, SVG, or PDF format.

   Click (expression options) next to an expression and then click any of the actions such as Edit (available only for guided expressions) or Duplicate.

   Click to modify chart configuration settings and then click Apply.

   Set the display_monitored_object_type system property to ‘true’ to configure Advanced Insights Explorer to show the specific monitored object type that is associated with a referenced source metric type.

   A single source metric type can be associated with multiple monitored object types, therefore it is useful when the specific monitored object type appears next to the source metric type that is referenced. For example, ‘% Idle Time’ is associated with ‘Microsoft.Linux.RHEL.5.OperatingSystem’ and ‘Microsoft.Windows.Server.2003.LogicalDisk’.

   - To open the System Property (sys_properties) table, enter sys_properties.list in the navigation filter.
   - To see the Monitored Object Type and Source Metric Type combinations in the Monitoring System Metric Types list view, navigate to Operational Intelligence > Configure > Metric Types.

Create a custom function

Create a custom function, which is a JavaScript that you can then use in a script expression in Advanced Insights Explorer.

Role required: evt_mgmt_user

1. Navigate to Operational Intelligence > Settings > Metric Expression Functions.
2. Click **New** and then fill out the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The function that the argument is associated with.</td>
</tr>
<tr>
<td>Eval</td>
<td>JavaScript associated with the function.</td>
</tr>
<tr>
<td>Order</td>
<td>The order in which the function is added to the environment. For example, if one function is calling another function, then the order of the first function is higher than the order of the second function.</td>
</tr>
<tr>
<td>Active</td>
<td>Select to denote that the function is active.</td>
</tr>
<tr>
<td>Return Type</td>
<td>Type of what the function returns: NUMBER, SERIES, or STRING.</td>
</tr>
</tbody>
</table>

For example, the script for a function named ‘foo’, would be similar to:

```javascript
var foo = function(bar) {
    //Your JS code here
}
```

3. Click **Submit**.

4. Navigate to **Operational Intelligence > Settings > Metric Expression Function Arguments** and create a record for each argument in the function:
   a) Click **New**.
   b) Fill out the form for the argument.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function Name</td>
<td>The function that the argument is associated with.</td>
</tr>
<tr>
<td>Order</td>
<td>Denotes the order of arguments in the function. For example, for a function <code>foo(arg1, arg2)</code>, two records for the two arguments are needed. In one record, <code>Order = 1</code>, and in the other <code>Order = 2</code>.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of the argument: NUMBER, SERIES, or STRING.</td>
</tr>
<tr>
<td>Domain</td>
<td>Domain associated with the argument.</td>
</tr>
</tbody>
</table>

c) Click **Submit**.

In Advanced Insights Explorer, you can now use this function in a script expression.

For example: `foo('testArg');`
Agent Workspace for Operational Intelligence

Agent Workspace for Operational Intelligence provides an easy-to-navigate interface where service agents can view the health of a CI associated with an alert. Health details for a CI include various metric charts with control bounds, and aggregations for a time period.

In Agent Workspace, when viewing an IT alert with a CI associated with it, click the Metrics button on the alert form to access the following Operational Intelligence details:

- Charts, control bounds, and aggregations for key metrics for CIs, in a single location.
- Aggregation of metrics for CIs with resources, in a single consolidated time series.
- Ability for administrators to configure which metric types appear for each CI class.

The set of metric charts that appear for a CI class in Agent Workspace, is determined by metric chart configurations in the Metric Dashboard Metadata (sa_metric_dashboard_metadata) table. The Metric Dashboard Metadata table is initially populated with configurations for common classes and important metrics for those classes. A user with the admin or evt_mgmt_admin role can update the default metric chart configurations in the Metric Dashboard Metadata table.

There can be only up to 10 active metric charts configured for a class at any point.

Plugins

All plugins related to Event Management, Operational Intelligence, and Agent Workspace, are required.

Roles

- To access Operational Intelligence metrics in Agent Workspace, the workspace_agent and itil roles are required.
- To configure the set of metrics that appears in Agent Workspace for a class, the admin or evt_mgmt_admin role is required.

Chart types

A metric chart that appears in the metric charts form in Agent Workspace can be a time-series chart or a single-score chart.

Time-series chart
The time-series chart type is a graph of all data points in the specified time range, for the metrics of the CI associated with the chart. Time-series charts are similar to charts in Insights Explorer, and depending on settings, show lower and upper bounds.

**Single-score chart**
The single-score chart type shows a single number which is an aggregation of the data points in the time series for the metric and CI of the chart. The aggregation is an average of the data points, or the maximum, minimum, or last data point in the time series.

View metric charts in Agent Workspace

Agent Workspace for Operational Intelligence provides easy access to metric charts for a CI. On an alert form, you can drill down to important metric charts for the CI that is associated with the alert.

- The CI that you want to view metric charts for, must be associated with an alert.
- The Metric Dashboard Metadata (sa_metric_dashboard_metadata) table must contain active chart configurations for the CI class for which you want to view metric charts.

Role required: workspace_agent and itil

The set of metric charts, chart types, and other chart characteristics, is determined by the chart configurations in the Metric Dashboard Metadata (sa_metric_dashboard_metadata) table, for the CI class.

1. Navigate to Agent Workspace and click Agent Workspace Home.
2. In Agent Workspace, click List and in the Lists panel, select one of the Alerts categorized lists such as All Alerts or Top Priority.
3. In the alerts list view, open an alert.
   The alert must have a Configuration item associated with it.
4. In the Details tab of the alert form, click Metrics.

All metric charts that are configured for the CI associated with the alert, and that are active, appear on the metric charts form in the Metrics tab. The calculation of the default time range for the charts is based on the times that the first and last events that generated the alert, occurred.
• Modify the time range for the charts by changing **Start Time** and **End Time**. When you modify the time range, all charts are updated to reflect the new time range.
• Click the cog wheel **Open settings** icon and change the settings for all charts:
  - **Show bounds**: Show the bounds in the chart. Even if enabled, there are no bounds in a chart for aggregated resources.
  - **Show legend**: Show chart legend.
• Click **Insights Explorer** to open the Insights Explorer in a new browser tab. That Insights Explorer opens with the current CI pre-selected and with the current time range settings in Agent Workspace.

**Configure metric charts for Agent Workspace**

Configure the set of Operational Intelligence metric charts, chart types, and other chart characteristics, that appear in Agent Workspace.

Role required: admin or evt_mgmt_admin

Up to 10 metric charts for a class can be configured to appear in the Operational Intelligence metric charts form in Agent Workspace. Metric chart configurations are stored in the Metric Dashboard Metadata (sa_metric_dashboard_metadata) table, determining which metric charts appear for a CI that is associated with an alert. By default, the Metric Dashboard Metadata contains pre-configured metric charts for several classes.

If **resource binding** is enabled and you specify a **Resource Type** in the metric chart configuration, then all the resources for the CI are included in the chart. The class table for the resource class (such as **ci_resource_metric_extension**) that you specify is used for retrieving all the resources for the CI. Then, for each point in time in the data series, the values of resources are aggregated into a single data point which is the average of the metric values of all resources for that point of time.

**Chart aggregation**

Aggregation is applied differently, depending on the setting of **Chart Type**, and on whether resources are included in the chart as indicated by the setting of **Resource Type**:

<table>
<thead>
<tr>
<th>Chart type</th>
<th>Resources</th>
<th>Aggregation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time_Series</td>
<td>No</td>
<td>No aggregation</td>
</tr>
<tr>
<td>Time_Series</td>
<td>Yes</td>
<td>Resources are averaged per data point.</td>
</tr>
<tr>
<td>Single_Score</td>
<td>No</td>
<td>Data points are aggregated according to the <strong>Chart Aggregate</strong> setting.</td>
</tr>
<tr>
<td>Single_Score</td>
<td>Yes</td>
<td>Resources are averaged per data point and then data points are aggregated according to the <strong>Chart Aggregate</strong> setting.</td>
</tr>
</tbody>
</table>

1. **Navigate to Operational Intelligence > Settings > Dashboard Metadata.**
2. In the Metric Dashboard Metadatas (Dashboard Metadata) list view, select an existing configuration or click **New**.
3. **Fill out the Metric Dashboard Metadata form.**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Include the chart on the Operational Intelligence charts form in Agent Workspace.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Resource class associated with the CI and for which to retrieve and aggregate all resources.</td>
</tr>
<tr>
<td>Configuration Item Type</td>
<td>CI class that the chart applies to such as cmdb_ci_metric_extension.</td>
</tr>
<tr>
<td>Chart Aggregate</td>
<td>Type of aggregation that is applied for single-score charts. Aggregation can be the average of the data points in the time series. Or, the minimum, maximum, or the last data point in the time series.</td>
</tr>
<tr>
<td>Source Metric Type</td>
<td>Metric such as “% Free Space” that is being monitored for the CI type.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which the metric chart appears within all metric charts specified for the class. For proper appearance and alignment, all single-score charts always appear after any time-series charts, regardless of order settings.</td>
</tr>
<tr>
<td>Chart Type</td>
<td>• Time_Series: Charts all the data points in the time series for the metric and CI of the chart.</td>
</tr>
<tr>
<td></td>
<td>• Single_Score: A single number which is the aggregation of all data points in the time series for the metric and CI of the chart. Aggregation is determined by the setting of Chart Aggregate.</td>
</tr>
</tbody>
</table>

**Self-health monitoring for Operational Intelligence**

Use Event Management self-health monitors to monitor the health of Operational Intelligence infrastructure components and processes, and to alert about potential issues. Self-health monitoring allows you to proactively remediate issues and minimize data loss.

Operational Intelligence is pre-configured to use Event Management self-health monitors to ensure that essential components and processes are functioning properly. If a monitor detects a failure, then the Event Management self-health monitoring system creates an event for the issue. These events are then converted into an Event Management alert or get appended to a previous alert. Operational Intelligence self-health monitors run every 60 seconds.

For example, if an Operational Intelligence MID Server distributed cluster experiences an issue, an event is generated with the status of **Major**. Then, this event is converted into an alert. Later, when the issue is resolved, a new event is generated with the status of **Info** and the original alert is closed.
Monitored components and processes

<table>
<thead>
<tr>
<th>Component/Process Check</th>
<th>Error Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operational Intelligence extension is running properly</td>
<td>Monitor checks the status field of the ecc_agent_ext_context_metric table to detect a state of Error or Warning.</td>
</tr>
<tr>
<td>Operational Intelligence MID Server distributed cluster is running properly</td>
<td>Monitor checks the status field of the ecc_agent_cluster table to detect a state of Error, Segmented, or Stopped.</td>
</tr>
<tr>
<td>Note: For a state of Stopped, an event is created only if the error_message field is not empty.</td>
<td></td>
</tr>
</tbody>
</table>
| Metrics are collected by active connectors that are configured for metrics collection | For each connector with the Metrics collection flag set to true:  
  Monitor checks the status of the last_kpi_status field in the em_connector_instance table to detect a status of Error. |
| Essential Operational Intelligence scheduled jobs are active  | Monitor checks the following essential scheduled jobs, to detect if any is inactive:  
  • Operational Intelligence — Metric configuration job  
  • Operational Intelligence — Sync tables with mid  
  • Operational Intelligence — Validate MID Distributed cluster |

For more information about enabling or disabling an Operational Intelligence self-health monitor, see Configure a self-health monitor.

View self-health monitoring statistics for Operational Intelligence

Monitor the status of components and processes of Operational Intelligence. Use the XMLStats page to view statistics and diagnostic details that can help with troubleshooting issues with Operational Intelligence.

1. Open a browser and enter the URL to the instance.  
   Type in: http://<instance name>.service-now.com
2. Append the following text to the instance URL: xmlstats.do?include=XMLStatsOI.  
   The complete URL format is: http://<instance name>.service-now.com/xmlstats.do?include=XMLStatsOI
3. Hit Enter.

Details are separated between two sections, instance level statistics and node level statistics. Details in the node section are specific to the node in the URL.

Node section

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>batches_recieved</td>
<td>Number of batches of metrics that the node received from the MID Server.</td>
</tr>
</tbody>
</table>
Details in the instance section are for the entire instance, showing the status of the self-health monitors.

**Instance section**

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>batches_processed</td>
<td>Number of batches of metrics processed on the node.</td>
</tr>
<tr>
<td>scheduled_jobs_monitor</td>
<td>Status of the self-health monitor that checks if the essential Operational Intelligence scheduled jobs are active.</td>
</tr>
<tr>
<td>distributed_cluster_monitor</td>
<td>Status of the self-health monitor that checks if the Operational Intelligence MID Server distributed cluster is running properly.</td>
</tr>
<tr>
<td>oi_extension_monitor</td>
<td>Status of the self-health monitor that checks if the Operational Intelligence extension is running properly.</td>
</tr>
<tr>
<td>metrics_collection_monitor</td>
<td>Status of the self-health monitor that checks if metrics are being collected by active connectors that are configured for metrics collection.</td>
</tr>
</tbody>
</table>

**Domain separation and Operational Intelligence**

This is an overview of domain separation and Operational Intelligence. Domain separation enables you to separate data, processes, and administrative tasks into logical groupings called domains. You can then control several aspects of this separation, including which users can see and access data.

**Overview**

**Support: Level 1**

Domain separation is supported in this application. Not all ServiceNow applications support domain separation; some include limitations on the data and administrative settings that can be domain separated. To learn more, see [Application support for domain separation](#).

**How domain separation works in Operational Intelligence**

When domains are separated in Operational Intelligence, users can view and manage alerts and events only in their own (tenant) domain. After the binding process has finished, the domain of the configuration item (CI) is used so that users within that domain can explore metric data. A domain column is present for Operational Intelligence tables that are provided in the base system. The domain column shows the name of the domain to which the CI belongs.

**Note:** If the domain column does not appear in the list, click (Personalize List) and add the required column. In addition, you can add a column that displays the domain path.
The following Operational Intelligence features have limited domain separation support.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric types</td>
<td>Supported at global level. The instance owner can control the metric types for all domains.</td>
</tr>
<tr>
<td>Metric type registration</td>
<td>Supported at global level. The instance owner can control the metric type registration for all domains.</td>
</tr>
<tr>
<td>Metric class</td>
<td>Supported at global level. The instance owner can control the metric classes that are used for all domains.</td>
</tr>
<tr>
<td>Configuration settings rules</td>
<td>Supported at global level. The instance owner can control the configuration settings rules for all domains.</td>
</tr>
</tbody>
</table>

Configure domain separation

You can configure Operational Intelligence for domain separation to create logically defined domains that limit unauthorized access to data.

**Before you begin**

Role required: evt_mgmt_admin and evt_mgmt_integration

**Procedure**

2. Configure the MID Server for Operational Intelligence as a user that belongs to the desired domain with the role `mid_server` assigned to it.
3. Configure a MID Server distributed cluster in the domain you desire and add Operational Intelligence MID Servers that belong to the same domain. For more information about setting up MID Servers for Operational Intelligence, see [MID Server and MID Server distributed cluster for Operational Intelligence](#).

**Note:** The Event Management’s binding events are configured using the domain of the logged-in MID Server user. Once the process has finished, the record’s domain is updated to use the domain from the CI so that any user within that domain can see the metric data.

**ITOM Optimization**

The ServiceNow® ITOM Optimization product includes two features: *Cloud Management* and *Cloud Insights*. 
Features of ITOM Optimization

Deployed together, the features provide an integrated, service-centric approach that maximizes business service quality, drives efficient processes, and ensures robust governance.

Cloud Management

The Cloud Management feature gives you tools to provision private and public cloud infrastructure and services and to achieve consistent management and cost visibility.

Cloud Insights

The Cloud Insights feature, available in the ServiceNow Store, helps you to analyze the full range of costs associated with cloud assets so you can identify and take action on opportunities to save money and optimize operations.

Using guided setup to implement IT Operations Management applications

IT Operations Management Guided Setup provides a sequence of tasks that help you configure IT Operations Management applications on your ServiceNow instance. To open IT Operations Management guided setup, navigate to Guided Setup > ITOM Guided Setup. For more information about using the guided setup interface, see .

Cloud Management

With the ServiceNow® Cloud Management application, you can use a single interface to access cloud resources, publish cloud offerings to a catalog, and manage the usage of those resources. You can also manage the life cycles of those resources. Cloud Management is integrated with both private and public cloud providers, including Amazon Web Services, Microsoft Azure, and VMware offerings.

Cloud Management is available as a separate subscription from the rest of the platform. You must also request activation from ServiceNow personnel.

Note:

This Cloud Management application is new starting with the Jakarta release. The new Cloud Management application is incompatible with the legacy Cloud Management application. The legacy and the new Cloud Management applications cannot be used simultaneously.

Explore
- Upgrade to New York
- Domain separation in Cloud Management

Set up
- Day 1 setup guide for Azure on Cloud Management
- Day 1 setup guide for Amazon Web Services on Cloud Management
- Day 1 setup guide for VMware on Cloud Management

Administer
- Cloud Management administration guide

Use
- Cloud User Portal

Develop
- Developer training

Troubleshoot and get help
Using guided setup to implement Cloud Management

Cloud Management guided setup provides a sequence of tasks that help you configure Cloud Management on your instance. To open Cloud Management guided setup, navigate to Guided Setup > ITOM Guided Setup.

For more information about using the guided setup interface, see Using guided setup.

Request the Cloud Management application

The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt).

Role required: admin

To purchase a subscription, contact your ServiceNow account manager. The account manager can arrange to have the plugin activated on your organization's production and sub-production instances, generally within a few days.

If you do not have an account manager, decide to delay activation after purchase, or want to evaluate the product on a sub-production instance without charge, follow these steps.

1. From your instance, navigate to System Definition > Plugins.
2. On the All Applications page, click Request Plugin to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>Date and time must be at least two business days from the current time.</td>
</tr>
<tr>
<td>Reason/Comments</td>
<td>Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments.</td>
</tr>
</tbody>
</table>

5. Click Submit.

Day 1 setup guide for Amazon Web Services on Cloud Management

To set up Cloud Management for the very first time, you perform the procedures in this "Day 1" setup guide. Be sure to perform the procedures in order. After you have performed Day 1 setup, you can perform optional Day 2 setup and configuration procedures as needed and in any order. Detailed instructions for each procedure follow this overview.

Request the Cloud Management application

The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt). See Request the Cloud Management application.

Roles required to set up Cloud Management

- Operations in the AWS Management Console require the administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.
About terms that Cloud Management uses

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In AWS, the region-specific containers for virtual resources are called regions. In Cloud Management, regions are called datacenters or logical datacenters (LDCs). The term “logical” is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.

Quick overview of the setup process

Your setup process includes these tasks:

- If needed: Request the Cloud Management application.
- Assign appropriate roles to cloud users.
- Set up the MID Servers that will handle secure communications with the provider API endpoints.
- On the provider portal, collect your account settings and the credentials that the Discovery process will use (through a MID Server) to programmatically access your provider accounts. Securely associate the account settings and credentials with a service account in Cloud Management.
- Set up a cloud account to represent your entire managed cloud infrastructure and set up a service account that works with one of your provider accounts. You specify which datacenters in the service account should be included in the cloud account. (Later, on “Day 2”, you can set up additional cloud accounts and service accounts from the same or other providers.)
- Cloud Management uses CMDB data to help users request and manage cloud resources and to help you manage your cloud infrastructure. To populate the CMDB with resource data for all datacenters, you manually run the Discovery process on each datacenter in the service account. Then, to ensure that the data continues to be updated, you configure Discovery to run on a regular schedule. Your cloud account might look like this:

![Cloud Account Diagram]

- Providers offer services that can auto-update the CMDB whenever a create/modify/terminate life-cycle change or configuration update occurs to a resource. You can configure the service to integrate directly with Cloud Management.
What you will do to integrate Cloud Management with your AWS accounts

Detailed instructions for each procedure follow this overview.

1. Assign roles to AWS users of Cloud Management

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

2. Set up MID Servers to connect Cloud Management to an AWS account

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks. You can configure the MID Server for Cloud Management to use a proxy server. Using a proxy server supports all cloud-based activities such as running Discovery, billing downloads, provisioning virtual machines, and running life-cycle operations on virtual machines.

**Note:** Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.

3. Create the credentials that enable Cloud Management to access your AWS data

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you first create a programmatic user in the AWS Management Console. You then securely store the credentials in a service account in your instance.

4. Create AWS GovCloud (US) credentials for Cloud Management

**Note:** Skip this procedure if your organization does not use AWS GovCloud (US).

To securely access data on your provider account, the Discovery process must present appropriate credentials. An AWS GovCloud (US) region is an isolated AWS region that meets stringent US government security and compliance requirements to host sensitive workloads. Cloud Management supports all AWS GovCloud (US) services.

5. Create a service account for AWS GovCloud (US)

**Note:** Skip this procedure if your organization does not use AWS GovCloud (US).

If your organization uses AWS GovCloud (US) regions, you create a service account for each region.

6. Set up a cloud account and service account for AWS

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter. A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

7. Set up AWS event processing for Discovery and Service Mapping
To integrate AWS events, configure a topic and enable event tracking in AWS.

8. Define the schedule for downloading AWS billing data

Define the scheduled job that regularly uses a MID Server to download billing data from the provider. Cloud Management saves the data in a cost table and uses the information to generate reports.

Next steps

When you have finished all Day-1 and Day-2 procedures in this setup guide, see the Cloud Management administration guide for information on using the Cloud Management application in your organization.

Assign roles to AWS users of Cloud Management

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

Role required: user_admin or admin

Assign the following roles to groups and users as appropriate:

<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root administrator [sn_cmp.cmp_root_admin]</td>
<td>Highest level of application access for Cloud Management.</td>
<td>All</td>
</tr>
<tr>
<td>Cloud administrator [sn_cmp.cloud_admin]</td>
<td>Configures the Cloud Management application and sets up the cloud infrastructure.</td>
<td>Cloud infrastructure: Service accounts and cloud account</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Networks and IPAMp</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Settings for provider services that auto-update the CMDB:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AWS Config</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Azure Alert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Google Stackdriver Logging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• IBM Cloud Update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• VMware Events</td>
</tr>
<tr>
<td>Governor [sn_cmp.cloud_governor]</td>
<td>Monitors overall cloud usage and enforces compliance of the organization’s rules,</td>
<td>Governance:</td>
</tr>
<tr>
<td></td>
<td>quotas, and policies. Also manages tags and permissions to various objects.</td>
<td>• Policies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Pools</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Quotas</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Permissions</td>
</tr>
<tr>
<td>Service Designer [sn_cmp.cloud_service_designer]</td>
<td>Creates blueprints, ARM and CloudFormation templates, and catalog items.</td>
<td>Cloud Service Design:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cloud templates</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Blueprints</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Blueprint catalog items</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Resource catalog items</td>
</tr>
</tbody>
</table>
### Descriptive name and role name

<table>
<thead>
<tr>
<th>Role Name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud user</td>
<td>Requests and manages stacks and resources.</td>
<td>Cloud User Portal</td>
</tr>
<tr>
<td>[sn_cmp.cloud_service_user]</td>
<td></td>
<td>When you assign the role to a group, all members of the group share quota limitations and ownership of certain resources.</td>
</tr>
<tr>
<td>Cloud operator</td>
<td>Monitors and troubleshoots the Cloud Management application.</td>
<td>Dashboards and reports:</td>
</tr>
<tr>
<td>[sn_cmp.cloud_operator]</td>
<td></td>
<td>- Cloud Operations Dashboard</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Cloud Root Cause Analysis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Cloud Orchestration Trail</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Cloud API Trail</td>
</tr>
<tr>
<td>Cloud Event Integration</td>
<td>Authorizes access to the instance for external services that auto-update the CMDB when cloud events occur. This role gives the access to the cloud event REST endpoint.</td>
<td>Cloud infrastructure:</td>
</tr>
<tr>
<td>[sn_cmp.cloud_event_integration]</td>
<td></td>
<td>- Service accounts and cloud account</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Networks and IPAM</td>
</tr>
<tr>
<td>Cloud Group administrator</td>
<td>Grants admin access to any group that you belong to.</td>
<td>Settings for provider services that auto-update the CMDB:</td>
</tr>
<tr>
<td>[sn_cmp.cloud_group_admin]</td>
<td></td>
<td>- AWS Config</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Azure Alert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Google Stackdriver Logging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- VMware Events</td>
</tr>
</tbody>
</table>

### Set up MID Servers to connect Cloud Management to an AWS account

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

Role required: admin or sn_cmp.cloud_admin

- Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.
- Internal network connection between the MID Servers and the AWS Cloud API endpoints: *.amazonaws.com
- You can configure the MID Server for Cloud Management to use a proxy server. See Using a proxy server for your MID Server.

1. You configure the connection to a MID Server in the same way as for any other applications that use MID Servers, like Discovery and Service Mapping. See MID Server for instructions on setting up and configuring MID Servers.
Note: Do not configure the MID Server Application related list.

2. Add the **Cloud Management** and **AWS** capabilities to the MID Server. See [Configure MID Server capabilities](#) for instructions. Alternatively, you can add the **ALL** capability, which includes the **Cloud Management** and the **AWS** capabilities.

Assuming member roles with an AWS API
The MID Server can call an AWS API and use the permanent credentials of an AWS master account (organization) to assume the role of one or more member accounts. By assuming the role, the MID Server receives temporary credentials for the member accounts generated by AWS for that role.

Assuming member account roles in a large AWS organization is more convenient and offers better security than using large numbers of permanent credentials for all member accounts. Temporary credentials are only acquired on behalf of a member account when there is no permanent credential specified for that member account in the Service Accounts (cmdb_ci_cloud_service_account) table.

AWS temporary credentials are used by Cloud Discovery to eliminate the need to manage AWS account credentials on your instance. For information on how to configure the MID Server to use these credentials, see [Assume an AWS role for temporary cloud Discovery credentials](#).

The MID Server uses the **AssumeRole** action in the **AWS Security Token Service API** to assume a member account role. Parameters passed to this API determine what additional security restrictions are applied to the role when it accesses AWS resources.

**Default member role configuration**
By default, the MID Server is configured to assume the **OrganizationAccountAccessRole**, which grants temporary credentials to all the members of a master account. This action occurs automatically if no permanent credentials exist for the member accounts. This configuration does not apply any additional security or restrict access to any resources in member accounts.

**Advanced member role configuration**
You can improve security by defining additional roles that a MID Server can assume.

Create records in the Cloud Management **AWS Org Assume Role Parameters** module that specify the roles and restrictions that apply. Records in the Cloud Service Account AWS Org Assume Role Params (cloud_service_account_aws_org_assume_role_params) table pass their parameters to the AWS Security Token Service API, which then provides the appropriate credentials and permissions to the MID Server.

**Note:** By default, temporary credentials for member accounts are cached for 60 minutes. For the MID Server properties used to control credential caching, see [Caching of temporary credentials](#).

Configure a custom AWS member role
Customize the AWS roles that a MID Server can assume to receive temporary credentials for member accounts. You can configure additional parameters to improve security and customize the way that the member account’s role is assumed when discovering cloud resources.

Role required: admin
Values that you enter in the Cloud Service Account AWS Org Assume Role Params (cloud_service_account_aws_org_assume_role_params) table are passed as parameters to the AWS AssumeRole API for the named service account.

1. Navigate to **Cloud Management > Organization Access Parameters > AWS Org Assume Role Parameters**.

2. Click **New** and then complete the form using these parameters:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access role name</td>
<td>Name of the AWS role in the member account which is used by the master account to acquire temporary credentials. <strong>Default</strong>: OrganizationAccountAccessRole</td>
</tr>
<tr>
<td>Role session name</td>
<td>Name for the session using the temporary security credentials that may help in distinguishing use of a role by a principal or purpose. This session name is visible in the AWS Cloud Trail logs. See <a href="#">Cloud API Trail</a> and the <a href="#">AWS documentation</a> on AWS Cloud Trail for details. <strong>Default</strong>: master_account_id__&lt;master account ID number&gt; An example of this is: master_account_id__321003876149.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Credential TTL in seconds | Time in seconds for the temporary security credentials to live. **Default**: Calculated as follows:  
1. Retrieve the value in the `mid.aws.sts.assume_role.credential_ttl_minutes` **MID Server property**.  
2. Constrain this value to be between 15 and 720 minutes. If the setting in the property is less than 15 minutes, the system enters 15 minutes. If the setting is greater than 720 minutes, the system enters 720 minutes.  
3. Convert the resulting value into seconds. |
| External ID | Unique identifier that might be required by the trust policy of the role being assumed. **Default**: `ServiceNow_MID_Server` |
| Session policy | IAM policy in JSON format that further restricts the permissions of the temporary security credentials beyond the role configured policy. (JSON in AWS policy language.) **Default**: Blank |
| MFA (multifactor authentication) | Serial number of the Multi-Factor Authentication (MFA) device (hardware or virtual) used to authenticate the master account. **Default**: Blank |
| MFA token code | Token code supplied by the MFA device (hardware or virtual) used to authenticate the master account. **Default**: Blank |
| Cloud service account | Required. Service account to associate with the access parameters that you pass to the AWS AssumeRole API. Enter an account ID, either a master account or a member account, from the Service Accounts (cmdb_ci_cloud_service_account) table. |

**Note:** For information on how the parameters are used and what they mean, see the [AWS documentation](https://aws.amazon.com) on the AWS Security Token Service API for the AssumeRole action.

**Using a proxy server for your MID Server**
You can configure the MID Server for Cloud Management to use a proxy server. Using a proxy server supports all cloud-based activities such as running Discovery, billing downloads, provisioning virtual machines, and running life-cycle operations on virtual machines.

**Proxy server limitations**
- Windows or Linux platforms only are supported.  
- The Google Cloud Console integration does not support downloading billing data.  
- VMware is not supported.  
- Remote PowerShell scripts cannot be executed.  
- Custom APIs may not work.
### Supported proxy server configurations for Cloud Management

<table>
<thead>
<tr>
<th>Authentication/No Authentication</th>
<th>Authentication type</th>
<th>Proxy Server type</th>
</tr>
</thead>
<tbody>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>NTLM</td>
<td>HTTP/HTTPS</td>
</tr>
</tbody>
</table>

### Qualified proxy server configurations for Cloud Management

#### Qualified configuration settings

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Operating system</th>
<th>Proxy server</th>
<th>Authentication mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration 1</td>
<td>Linux</td>
<td>None</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Configuration 2</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>None</td>
</tr>
<tr>
<td>Configuration 3</td>
<td>Linux</td>
<td>Squid (HTTPS)</td>
<td>Local</td>
</tr>
<tr>
<td>Configuration 4</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>Active Directory</td>
</tr>
</tbody>
</table>

#### Qualified configurations

<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS, Azure, and IBM Cloud</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Google Cloud Platform</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Cloud provider</td>
<td>Functionality</td>
<td>Configuration 1</td>
<td>Configuration 2</td>
<td>Configuration 3</td>
<td>Configuration 4</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>-----------------</td>
</tr>
<tr>
<td></td>
<td>Billing</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>download</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scripts using</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>SSH</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scripts using</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>PowerShell</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VMware</td>
<td>Not Applicable</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Create AWS GovCloud (US) credentials for Cloud Management

Skip this procedure if your organization does not use AWS GovCloud (US). To securely access data on your provider account, the Discovery process must present appropriate credentials. An AWS GovCloud (US) region is an isolated AWS region that meets stringent US government security and compliance requirements to host sensitive workloads. Cloud Management supports all AWS GovCloud (US) services.

Note: Skip this procedure if your organization does not use AWS GovCloud (US).

- role required: sn_cmp.cloud_admin
- AWS GovCloud (US) access key ID and the secret access key that you generated on the AWS Management Console.

Cloud Management accesses GovCloud regions using a set of credentials for each region. To configure Cloud Management to support a GovCloud region, you create one standard AWS account for each region (required for billing), obtain the credentials for the account, and then create a service account for the region. For more information on billing, see the ‘AWS GovCloud (US) Billing and Payment’ page on AWS documentation.

1. On the AWS Management Console, enter IAM in the AWS services search box to open the Identity and Access Managements (IAM) service.
2. On the IAM Resources portal, click Users.
3. Create the Discovery user that will have programmatic access to your AWS GovCloud (US) resource and billing data. Click Add user. On the Details page, configure the user settings, and then click Next.

<table>
<thead>
<tr>
<th>User name</th>
<th>Name for the programmatic user, for example, servicenowcloud.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access type</td>
<td>Select Programmatic access.</td>
</tr>
</tbody>
</table>

4. On the Permissions page, you attach the user to a policy. Configure the following settings and then click Next.

| Set permissions for <username> | Select Attach existing policies directly. |
5. On the Review page, verify your selections and then click Create user.

6. On the Complete page, perform two steps:

   - **Note:** Do not leave the page until you have completed both steps. The Secret access key value will not appear again. You will paste the values that you generate in these steps into a Cloud Management form.

   1. Click **Show** to display the Secret access key. Copy the value.
   2. Click **Download .csv** to save the CSV-format file that contains the user name, Access key ID, and the Secret access key value. You create the file as a backup in the case that you lose the values. Verify that the file was created and then store the file securely.

7. On your instance, navigate to **Orchestration > Credentials & Connections > Credentials**.

8. Click **New**, select **AWS Credentials**, enter a unique and meaningful **Name** (for example, **AWS GovCloud Creds 01**), and then fill in the form.

<table>
<thead>
<tr>
<th>AWS Credentials form fields</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>Enter a unique and descriptive name for the AWS credentials.</td>
</tr>
<tr>
<td><strong>Active</strong></td>
<td>Select the check box to use the credential.</td>
</tr>
<tr>
<td><strong>Access Key ID</strong></td>
<td>Enter the <strong>Access key ID</strong> that you generated on the AWS Management Console, such as: <strong>APIAIOSFODNN7EXAMPLE</strong>.</td>
</tr>
<tr>
<td><strong>Secret Access Key</strong></td>
<td>Enter the <strong>Secret access key</strong> that you generated on the AWS Management Console, such as: <strong>wPalrXUtHFEH/K7MDENG/bPxRfICYEXAMPLEKEY</strong>.</td>
</tr>
</tbody>
</table>

9. Click **Submit**.

   The instance records the credentials. Next, you **Create a service account for AWS GovCloud (US)**.
Create a service account for AWS GovCloud (US)
If your organization uses AWS GovCloud (US) regions, you create a service account for each region. Skip this procedure if your organization does not use AWS GovCloud (US).

**Note:** Skip this procedure if your organization does not use AWS GovCloud (US).

- Role required: sn_cmp.cloud_admin
- AWS GovCloud (US) credentials for each GovCloud region.
- AWS GovCloud (US) account ID (from the AWS Management Console).

A service account holds the credential and account information that you created in your provider account. Discovery uses the information to access your provider account and then obtain information on each logical datacenter that is associated with the account.

1. Navigate to **Cloud Management > Service Accounts**.
2. Click **New**, enter a unique and descriptive name for the account (for example, **AWS GovCloud SA O1**) and then fill in the Cloud Service Account form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account ID</td>
<td>Account ID to which this credential belongs.</td>
</tr>
<tr>
<td>Discovery credentials</td>
<td>Select the name of the credentials that you created in the Create AWS GovCloud (US) credentials for Cloud Management procedure. In the example, you used the name AWS GovCloud Creds O1.</td>
</tr>
<tr>
<td>Datacenter URL</td>
<td>URL of the datacenter.</td>
</tr>
<tr>
<td>Datacenter type</td>
<td>Select the CMDB table that represents the type of datacenter.</td>
</tr>
<tr>
<td>Datacenter discovery status</td>
<td>Auto-generated value; Status and timestamp of the last execution of Discovery on the datacenter.</td>
</tr>
</tbody>
</table>

3. Click **Update** or **Submit**.
   The system creates the service account and displays the list of all discovered datacenters.

Repeat the process to create additional cloud accounts as needed. Run Discovery and set capacity limits for each datacenter.

Create the credentials that enable Cloud Management to access your AWS data
To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you first create a programmatic user in the AWS Management Console. You then securely store the credentials in a service account in your instance.

Roles required:
- AWS Management Console administrator
- Cloud Management: admin or sn_cmp.cloud_admin

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings.

1. On the AWS Management Console, enter **IAM** in the **AWS services** search box to open the Identity and Access Managements (IAM) service.
2. On the IAM Resources portal, click **Users**.

3. Create the Discovery user that will have programmatic access to your AWS resource and billing data. Click **Add user**. On the Details page, configure the user settings, and then click **Next**.

<table>
<thead>
<tr>
<th>User name</th>
<th>Name for the programmatic user, for example, servicenowcloud.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access type</td>
<td>Select <strong>Programmatic access</strong>.</td>
</tr>
</tbody>
</table>

4. On the Permissions page, you attach the user to a policy. Configure the following settings and then click **Next**.

<table>
<thead>
<tr>
<th>Set permissions for &lt;username&gt;</th>
<th>Select <strong>Attach existing policies directly</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attach one or more policies</td>
<td>Select the appropriate policy.</td>
</tr>
</tbody>
</table>

   **Note:** The **AdministratorAccess** policy has the most powerful permission level, including permission to provision cloud resources. The policy enables the same access that would be granted to the instance if you were not using IAM and used your AWS account Access Key ID and Secret Access Key. You might instead prefer to create a policy or combine multiple policies to grant the appropriate permission level. See [Create an IAM user policy for Cloud Management](#) for details.

5. On the Review page, verify your selections and then click **Create user**.

6. On the Complete page, perform two steps:

   **Note:** Do not leave the page until you have completed both steps. The **Secret access key** value will not appear again. You will paste the values that you generate in these steps into a Cloud Management form.

   1. Click **Show** to display the Secret access key. Copy the value.
   2. Click **Download .csv** to save the CSV-format file that contains the user name, Access key ID, and the Secret access key value. You create the file as a backup in the case that you lose the values. Verify that the file was created and then store the file securely.

7. On your instance, navigate to **Orchestration > Credentials & Connections > Credentials**.

8. Click **New**, select **AWS Credentials**, enter a unique and meaningful **Name** (for example, Cloud Management Account), and then fill in the form.

   **AWS Credentials form fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for the AWS credentials.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to use the credential.</td>
</tr>
<tr>
<td>Field</td>
<td>Input value</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Access Key ID</td>
<td>Enter the <strong>Access key ID</strong> that you generated on the AWS Management Console, such as: APIAIOSFODNN7EXMPLE.</td>
</tr>
<tr>
<td>Secret Access Key</td>
<td>Enter the <strong>Secret access key</strong> that you generated on the AWS Management Console, such as: wPairXUtFEMI/K7MDENG/bPxRfICYEXMEMEKEY.</td>
</tr>
</tbody>
</table>

9. Click **Update** or **Submit**.

Create an IAM user policy for Cloud Management
If you are using IAM to manage users in AWS, you must create a user profile in IAM that Discovery uses to access AWS data.

Familiarize yourself with the [AWS documentation](#) on IAM. You must know how to create an IAM user and set up a user policy.

1. Log in to the AWS Management Console and create a new user in IAM.
   The access key must be auto-generated. You need the key when you configure AWS credentials in the instance.
2. Save the Access Key ID and Secret Access Key.
3. Open the user record in the instance for appropriate user.
4. Define a user policy in AWS using either of the following methods:
   - Grant Administrator Access to the instance, which is essentially the same access that would be granted to the instance if you were not using IAM and used your AWS account Access Key ID and Secret Access Key. Attach the **AdministratorAccess** policy to the user profile.
     
     **Note:** To create a user policy that supports only Discovery rather than the provisioning of cloud resources, attach the **ReadOnlyAccess** policy instead.
   - Create a custom policy with a descriptive name and the following code in the Policy Document field in the user policy:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": "cloudfront:*",
    "Effect": "Allow",
    "Resource": "*"
  }, {
    "Action": "s3:*",
    "Effect": "Allow",
    "Resource": "arn:aws:s3:::*"
  }, {
    "Action": "elasticloadbalancing:*",
    "Effect": "Allow",
    "Resource": "*"
  }, {
    "Action": "sqs:*",
    "Effect": "Allow",
    "Resource": "*"
  }, {
    "Action": "rds:*",
    "Effect": "Allow",
    "Resource": "*"
  }]
```
},
{ "Action": "sns:*",
  "Effect": "Allow",
  "Resource": "*"
},
{ "Action": "ec2:*",
  "Effect": "Allow",
  "Resource": "*"
},
{ "Action": "cloudformation:*",
  "Effect": "Allow",
  "Resource": "*"
},
{ "Action": "directconnect:*",
  "Effect": "Allow",
  "Resource": "*"
},
{ "Action": "route53:*",
  "Effect": "Allow",
  "Resource": "arn:aws:route53:::*"
},
{ "Action": "dynamodb:*",
  "Effect": "Allow",
  "Resource": "*"
},
{ "Action": "apigateway:*",
  "Effect": "Allow",
  "Resource": "*"
},
{ "Action": "lambda:*",
  "Effect": "Allow",
  "Resource": "*"
},

{ "Action": [ "iam:DeleteServerCertificate",
              "iam:GetServerCertificate",
              "iam:ListServerCertificates",
              "iam:UpdateServerCertificate",
              "iam:UploadServerCertificate" ],
  "Effect": "Allow",
  "Resource": "arn:aws:iam::*:server-certificate/*"
},

{ "Action": [ "rds:DescribeDBInstances"
              ],
  "Effect": "Allow",
  "Resource": "*"
},

{ "Action": [ "elasticloadbalancing:DescribeLoadBalancers"
              ],
  "Effect": "Allow",
  "Resource": "*"
},

{ "Action": [ "dynamodb:ListTables"
              ],
  "Effect": "Allow",
  "Resource": "*"
},

{ "Action": [ "lambda:ListFunctions"
              ],
  "Effect": "Allow",
  "Resource": "*" }
Set up a cloud account and service account for AWS

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter. A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

- Operations in the AWS Management Console require the administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.

When you finish the Day-1 setup process, your cloud account might look like this:

```json
{
    "Action": [ "apigateway:List*", "*" ],
    "Effect": "Allow",
    "Resource": "*"
}
{
    "Effect": "Allow",
    "Resource": "*"
}
}```

2. Select or create the cloud account:
   - Select an existing cloud account:
     1. On the Cloud Account page, click **Configure**.
     2. On the General Information page, click **Next** and then continue with the next step.
   - Create a cloud account:
     1. On the General Information page, click **New** and then enter a unique and meaningful **Name** and **Description** for the cloud account.
     2. Select the provider.
     3. Click **Next** and then continue with the next step.

3. On the Datacenters page, create a Service Account: Click the + next to the **Service Account** field, enter a unique and meaningful **Name** for the service account and then fill in the form.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account ID</td>
<td>12-digit IAM user account number. Expand the list under the AWS account name on the AWS Management Console to view the number. <strong>Note:</strong> In the <strong>Account ID</strong> field, remove the dash characters from the number.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Discovery credentials        | • Select the name of the credentials that you created in the Create the credentials that enable Cloud Management to access your AWS data procedure. In the example, you used the name AWS Main Account.  
  • Leave the field blank if the account is a member account of an AWS Organization and you have configured the associated master account with a credential.       |
| Datacenter URL               | Required only for AWS GovCloud (US) accounts: Full URL of the datacenter, for example: https://<IP-of-center>/sdk.                                                                                             |
| Datacenter type              | Select AWS datacenter.                                                                                                                                                                                      |
| Datacenter discovery status  | Auto-generated value: Status and timestamp of the last execution of Discovery on the datacenter.                                                                                                            |
| Parent account               | Appears when you select AWS datacenter.  
  • Select the AWS master account that represents the Organization in AWS that this member account belongs to.  
  • Leave the field empty if this account is not part of an AWS Organization.                                                                                                                                   |
| Is master account            | Appears when you select AWS datacenter.  
  • Clear the check box if the account is not the Organization account.  
  • Select the check box to associate the AWS service account with the master account. The master account corresponds to the Organization in AWS. You can identify any service account in your instance as a master account only if you already configured the account in AWS as an Organization and you already associated other accounts under the Organization. See the AWS documentation for information on Organizations. |
| Should pull events           | Enables Cloud Management to collect events from resource types in AWS LDCs in the `sn_cmp_cloud_event_list`.                                                                                               |

4. Click **Submit** to create the service account.  
5. Click **Discover Datacenters**.  
6. When the datacenters appear, select one or more datacenters to include in the cloud account and then click **Save**. The Cloud Account dashboard appears. The datacenters that you selected appear on the **Datacenters** tab.
7. Click a datacenter.
   The following lists appear:
   - **Resources**: Cloud resources for the current datacenter. Run Discovery to populate the
     CMDB for the datacenter and populate the tab. See [Discover all datacenters in a service
     account on-demand](#).
   - **Capacity Limits**: Limits on virtual CPUs, virtual networks, storage volume size, and other
     services. See [Set capacity limits on user requests for resources](#).
   - **Discovery Log**: The process creates a log record for each action associated with a
     discovery status.

After you complete all procedures in this initial ‘Day 1’ setup, you can create additional cloud
accounts and service accounts to organize and compartmentalize your cloud infrastructure.

**Configure the Amazon AWS Config service to auto-update the CMDB**

Integrate AWS Config with a Now Platform instance to receive near real-time Simple Notification
Service (SNS) notifications from AWS. To process AWS events, you must configure AWS Config and
SNS on the Amazon console. The AWS Config service can auto-update CI data in the CMDB
whenever Cloud Management or your AWS account makes a life-cycle state or configuration
change to an Amazon resource. As a result, the CI data in the CMDB is updated without having to
wait for Discovery to run.

- **Role required**: For Cloud Management: sn_cmp.cloud_event_integration. For Discovery:
  discovery_admin.
- You need an account with Amazon subscriptions (service accounts) and associated logical
datacenters.
- The AWS Config role should have access to the SNS and AWS Config Service, and the resource
types you are recording configuration changes for.
- If authentication to cloud event endpoint is required, assign a user with the
  sn_cmp.cloud_event_integration role. This role is required to set up an Alert Rule in the
  instance.

When AWS Config sends an update, the instance processes the event and creates or updates the
CI entry in the CMDB and the CI information in the Cloud User Portal. Each event is saved as a
record in the Cloud Event (sn_cmp_cloud_event) table.

Many of the steps in the topic are performed in the AWS portal. If you want to review the AWS
documentation, see the following:

- [Set Up Amazon SNS Notifications](#)
- [Viewing the AWS Config dashboard](#)

1. Log in to your AWS account.
2. On the **Services** page, navigate to **AWS Services > Find Services > Simple Notification Service**.
3. Create a topic (the communications channel used to send messages and subscribe to
   notifications). On the **SNS dashboard**, click **Topics**.
   1. Click **Create new topic**.
   2. Enter the **Name** and **Display name - optional** for the topic and then click **Create topic**.
4. Create one or more subscriptions to the topic. Subscriptions subscribe endpoints to the topic
to receive messages that AWS Config publishes to the topic.
   1. On the **Topic Details** page, click **Create subscription** and then specify the following
      values:
### Protocol

For the subscription that auto-update the CMDB, specify **HTTPS**.

**Note:** When you have created the HTTPS subscription, you can create a separate subscription that sends email to a specified person.

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>URL of the instance plus additional information in the following format:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>https://&lt;username&gt;:&lt;user_password&gt;@&lt;instance_URL&gt;/api/now/cloud_event</code></td>
</tr>
</tbody>
</table>

**Example:**

```
https://abeltuter:123456@myinstance.service-now.com/api/now/cloud_event
```

2. Click **Create subscription**. Repeat the process to create as many subscriptions as needed.

5. On the **Services** page, navigate to **Config > Settings**.

6. On the **Settings** page, click **Turn on** to turn on recording.

This setting records the updates that happen to resources.

7. In the **Resource types to record** section: As needed, select **Record all resources supported in this region** or **Include global resources** (specify only the resource types for which CI classes exist in the system).

**Note:** Any configuration changes on the resources you have selected in the **Resource types to record** section, is considered an event and auto-updated to the CMDB.

8. Optional: To set a value other than the default data retention period, specify a period using the **Set a custom Data retention period for configuration items recorded by AWS Config** check box.

9. In the **Amazon S3 Bucket** section, **Create a bucket** or **Choose a bucket from your account** and choose the appropriate bucket from the **Bucket name** list.

10. In the **Amazon SNS topic** section, select **Stream configuration changes and notifications to an Amazon SNS topic** and then select **Choose a topic from your account**.

11. Select the **Topic name** from the list. AWS Config sends the updates to the topic.

12. In the **AWS Config role** section, select **Choose a role from your account**.

   a) Select a role, from the **Role name** list.

13. Click **Save**.

**Note:** If you have any trouble navigating the AWS interface, see Amazon documentation: [Selecting Which Resources AWS Config Records](#).

You have configured the AWS services to auto-update the CMDB.

After some events are generated, navigate to the **Cloud User Portal** to view the events.
Define the schedule for downloading AWS billing data

Define the scheduled job that regularly uses a MID Server to download billing data from the provider. Cloud Management saves the data in a cost table and uses the information to generate reports.

Role required: sn_cmp.cloud.governor

Prepare the following items before you begin:

- Set up a payer account that gets billing information for all linked accounts.
- Create the Amazon S3 bucket that is the data storage area for the daily detailed billing records on the AWS account. The credential that is used to access the report must have permissions to the S3 bucket. You must set the S3 bucket based on the Order location. The AssumeRole policies MUST include an S3 access policy. You might assume that the AWS account that you always use can create an S3 bucket in the region where the CFT is being provisioned. If you use STS-based authentication, however, the AWS account might not be able to create the bucket because authentication is tied to the AssumeRole policies. See the "Setting Up an Amazon S3 Bucket for AWS Cost and Usage Reports" page on the AWS documentation for details.
- Enable detailed billing reports on AWS. See the AWS documentation for details.

This procedure involves two tasks:

1. Fill in the AWS Billing Schedule form with information that you copy from the AWS portal. This action enables the instance to download AWS billing data.
2. Specify the schedule for downloading AWS billing information.

1. In the Cloud Admin Portal, navigate to Analyze > Billing.
2. On the Billing Schedules tab, click New, enter a unique and meaningful Name, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider</td>
<td>Select AWS.</td>
</tr>
<tr>
<td>Service Account</td>
<td>The AWS service account that you created during initial setup. (The AWS account to run the billing report on.)</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Bucket</td>
<td>Data storage area for the daily billing records on the AWS account. Follow this procedure to obtain the value:</td>
</tr>
<tr>
<td></td>
<td>1. Log in to the AWS Management Console and select My Billing Dashboard in the menu.</td>
</tr>
<tr>
<td></td>
<td>2. Select Reports.</td>
</tr>
<tr>
<td></td>
<td>3. Copy the value from the S3 bucket column into the Bucket field on the form.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to activate the download job.</td>
</tr>
<tr>
<td>Run</td>
<td>Frequency that the download job should run.</td>
</tr>
<tr>
<td>Time</td>
<td>Time of day that the download job should run.</td>
</tr>
</tbody>
</table>

3. Click **Save**. To run the schedule on demand, click **Execute Now**. On the Retrieve Billing Data window that appears, select a date range and then click **Download**.

*View cloud costs on the Billing dashboard*

Optional: Configure the download size of AWS billing data

A single set of cloud billing data can be large. The MID Server, therefore, sends the data to the ECC queue in manageable chunks. You can configure a system property to limit the size of each chunk to avoid performance issues caused by large data transfers.

Role required: admin

1. Open the System properties table by entering `sys_properties.list` in the application filter.
2. Filter the list to open the `sn_cmp.billing.page_size` property.
The default value is 3 MB.
3. Modify the value as needed. The maximum is 5 MB.
4. Save the property.

(Optional) Add an AWS service account to the cloud account

During Cloud Management Day 1 setup, you added one service account to the cloud account. To compartmentalize your infrastructure or to include different datacenters, you can add another service account. A particular datacenter, however, cannot be selected in more than one service account in a cloud account.

Role required: sn_cmp.cloud_admin

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter.

In this example, you added the service account named **ProviderB-ServiceAccount-1** and selected three datacenters to include in the cloud account:
**Note:** In a cloud account, you cannot select a particular datacenter in two different service accounts.

1. Navigate to **Cloud Management > Service Accounts**.
2. Click **New**, enter a unique and meaningful **Name**, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Account ID</strong></td>
<td>12-digit IAM user account number. Expand the list under the AWS account name on the AWS Management Console to view the number.</td>
</tr>
</tbody>
</table>

**Note:** In the **Account ID** field, remove the dash characters from the number.

![IAM User: MyUserName @ 1234-5678-9012](attachment:image.png)

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Discovery credentials | - Select the name of the credentials that you created in the [Create the credentials that enable Cloud Management to access your AWS data](https://servicecloud.com) procedure. In the example, you used the name **AWS Main Account**.  
  - Leave the field blank if the account is a member account of an AWS Organization and you have configured the associated master account with a credential. |
| Datacenter URL        | Required only for AWS GovCloud (US) accounts: Full URL of the datacenter, for example: `https://<IP-of-center>/sdk`.                                                                                       |
| Datacenter type       | Select **AWS datacenter**.                                                                                                                                                                                  |
| Datacenter discovery status | Auto-generated value: Status and timestamp of the last execution of Discovery on the datacenter.                                                                                                   |
| Parent account        | Appears when you select **AWS datacenter**.  
  - Select the AWS master account that represents the Organization in AWS that this member account belongs to.  
  - Leave the field empty if this account is not part of an AWS Organization. |
| Is master account     | Appears when you select **AWS datacenter**.  
  - Clear the check box if the account is not the Organization account.  
  - Select the check box to associate the AWS service account with the master account. The master account corresponds to the Organization in AWS.  
  You can identify any service account in your instance as a master account only if you already configured the account in AWS as an Organization and you already associated other accounts under the Organization. See the [AWS documentation](https://aws.amazon.com) for information on Organizations. |
| Should pull events    | Enables Cloud Management to collect events from resource types in AWS LDCs in the `sn_cmp_cloud_event_list`.                                                                                         |

3. **Click Update** or **Submit**.  
The system creates the service account and displays the list of all discovered datacenters.  
4. Repeat the process to add as many service accounts as needed.
Day 1 setup guide for Azure on Cloud Management

To set up Cloud Management for the very first time, you perform the procedures in this "Day 1" setup guide. Be sure to perform the procedures in order. After you have performed Day 1 setup, you can perform optional Day 2 setup and configuration procedures as needed and in any order. Detailed instructions for each procedure follow this overview.

Request the Cloud Management application

The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt). See Request the Cloud Management application.

Roles required to set up Cloud Management

- Operations on the Azure portal require the Active Directory administrator or Azure administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.

About terms that Cloud Management uses

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In Azure, the region-specific containers for virtual resources are called regions. In Cloud Management, regions are called datacenters or logical datacenters (LDCs). The term "logical" is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.

Quick overview of the setup process

Your setup process includes these tasks:

- If needed: Request the Cloud Management application.
- Assign appropriate roles to cloud users.
- Set up the MID Servers that will handle secure communications with the provider API endpoints.
- On the provider portal, collect your account settings and the credentials that the Discovery process will use (through a MID Server) to programatically access your provider accounts. Securely associate the account settings and credentials with a service account in Cloud Management.
- Set up a cloud account to represent your entire managed cloud infrastructure and set up a service account that works with one of your provider accounts. You specify which datacenters in the service account should be included in the cloud account. (Later, on "Day 2", you can set up additional cloud accounts and service accounts from the same or other providers.)
- Cloud Management uses CMDB data to help users request and manage cloud resources and to help you manage your cloud infrastructure. To populate the CMDB with resource data for all datacenters, you manually run the Discovery process on each datacenter in the service account. Then, to ensure that the data continues to be updated, you configure Discovery to run on a regular schedule. Your cloud account might look like this:
Providers offer services that can auto-update the CMDB whenever a create/modify/terminate life-cycle change or configuration update occurs to a resource. You can configure the service to integrate directly with Cloud Management.

What you will do to integrate Cloud Management with your Azure accounts

Detailed instructions for each procedure follow this overview.

1. Assign roles to Azure users of Cloud Management

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

2. Set up MID Servers to connect Cloud Management to an Azure account

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

Note: Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.

3. Create an Azure service principal

To securely access resource and billing data on your Azure account, the Discovery process must present appropriate Azure account credentials. You create a special programmatic account — an Azure service principal — to generate the required credentials.

4. Store the Azure service principal credentials in the instance

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you first create Azure service principal credentials in the Azure Portal. You then securely store the credentials in a service account in your instance.

5. Add an Azure service account
A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified data center.

6. Discover all datacenters in a service account on-demand

Discovery uses the information in the service account to identify all logical datacenters associated with the provider account.

7. Set up a cloud account for Azure

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

8. Set capacity limits on user requests for resources

Capacity limits place restrictions on the attributes of cloud resources such as the number of virtual machines, virtual CPUs, or aggregate storage. You can set limits on resources separately for each logical datacenter in a cloud account.

9. Configure the Azure Alert service to auto-update the CMDB

The Azure Alert service can auto-update CI data in the CMDB whenever Cloud Management or your Azure account makes a life-cycle state or configuration change to an Azure resource. As a result, the CI data in the CMDB is updated without having to wait for Discovery to run.

10. Define the schedule for downloading Azure billing data

Define the scheduled job that regularly uses a MID Server to download billing data from the provider. Cloud Management saves the data in a cost table and uses the information to generate reports.

Next steps

When you have finished all Day-1 and Day-2 procedures in this setup guide, see the Cloud Management administration guide for information on using the Cloud Management application in your organization.

Assign roles to Azure users of Cloud Management

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

Role required: user_admin or admin

Assign the following roles to groups and users as appropriate:

<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root administrator [sn_cmp.cmp_root_admin]</td>
<td>Highest level of application access for Cloud Management.</td>
<td>All</td>
</tr>
<tr>
<td>Descriptive name and role name</td>
<td>Description and tasks</td>
<td>Access rights in Cloud Management</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td><strong>Cloud administrator</strong> [sn_cmp.cloud_admin]</td>
<td>Configures the Cloud Management application and sets up the cloud infrastructure.</td>
<td>Cloud infrastructure: Service accounts and cloud account, Networks and IPAM, Settings for provider services that auto-update the CMDB: - AWS Config - Azure Alert - Google Stackdriver Logging - IBM Cloud Update - VMware Events</td>
</tr>
<tr>
<td><strong>Governor</strong> [sn_cmp.cloud_governor]</td>
<td>Monitors overall cloud usage and enforces compliance of the organization’s rules, quotas, and policies. Also manages tags and permissions to various objects.</td>
<td>Governance: - Policies - Pools - Quotas - Permissions</td>
</tr>
<tr>
<td><strong>Service Designer</strong> [sn_cmp.cloud_service_designer]</td>
<td>Creates blueprints, ARM and CloudFormation templates, and catalog items.</td>
<td>Cloud Service Design: - Cloud templates - Blueprints - Blueprint catalog items - Resource blocks</td>
</tr>
<tr>
<td><strong>Cloud user</strong> [sn_cmp.cloud_service_user]</td>
<td>Requests and manages stacks and resources.</td>
<td>Cloud User Portal When you assign the role to a group, all members of the group share quota limitations and ownership of certain resources.</td>
</tr>
<tr>
<td><strong>Cloud operator</strong> [sn_cmp.cloud_operator]</td>
<td>Monitors and troubleshoots the Cloud Management application.</td>
<td>Dashboards and reports: - Cloud Operations Dashboard - Cloud Root Cause Analysis - Cloud Orchestration Trail - Cloud API Trail</td>
</tr>
</tbody>
</table>
### Set up MID Servers to connect Cloud Management to an Azure account

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

Role required: admin or sn_cmp.cloud_admin

- Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.
- Internal network connection between the MID Servers and the Azure Cloud API endpoints: management.azure.com
- You can configure the MID Server for Cloud Management to use a proxy server. See *Using a proxy server for your MID Server*.

You configure the connection to a MID Server in the same way as for any other applications that use MID Servers, like Discovery and Service Mapping. See *MID Server* for instructions on setting up and configuring MID Servers.

**Note:** Do not configure the MID Server Application related list.

### Using a proxy server for your MID Server

You can configure the MID Server for Cloud Management to use a proxy server. Using a proxy server supports all cloud-based activities such as running Discovery, billing downloads, provisioning virtual machines, and running life-cycle operations on virtual machines.

**Proxy server limitations**

- Windows or Linux platforms only are supported.
- The Google Cloud Console integration does not support downloading billing data.
- VMware is not supported.
- Remote PowerShell scripts cannot be executed.
- Custom APIs may not work.

Supported proxy server configurations for Cloud Management

<table>
<thead>
<tr>
<th>Authentication/No Authentication</th>
<th>Authentication type</th>
<th>Proxy Server type</th>
</tr>
</thead>
<tbody>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>NTLM</td>
<td>HTTP/HTTPS</td>
</tr>
</tbody>
</table>

Qualified proxy server configurations for Cloud Management

Qualified configuration settings

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Operating system</th>
<th>Proxy server</th>
<th>Authentication mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration 1</td>
<td>Linux</td>
<td>None</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Configuration 2</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>None</td>
</tr>
<tr>
<td>Configuration 3</td>
<td>Linux</td>
<td>Squid (HTTPS)</td>
<td>Local</td>
</tr>
<tr>
<td>Configuration 4</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>Active Directory</td>
</tr>
</tbody>
</table>

Qualified configurations

<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS, Azure,</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>and IBM Cloud</td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>PowerShell</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Google Cloud</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Platform</td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Create an Azure service principal

To securely access resource and billing data on your Azure account, the Discovery process must present appropriate Azure account credentials. You create a special programmatic account — an Azure service principal — to generate the required credentials.

On the Azure Portal, the Active Directory administrator or Azure administrator role is required.

A service principal for Azure cloud services is analogous to a Microsoft Windows service account that enables Windows processes to communicate with each other within an Active Directory domain.

You will enter the service principal credential values to create a service account in Cloud Management.

Note: Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. To ensure that you paste the correct Azure values into the Cloud Management forms, you will temporarily store the Azure values in a text file and then paste the values into the Cloud Management forms.

The text file that you generate during this procedure might look something like this example:
1. Open a text editor, paste the following text into the editor, and then save the file as Azure-Credentials.txt.

<table>
<thead>
<tr>
<th>Azure label</th>
<th>ServiceNow label</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directory ID</td>
<td>Tenant ID</td>
<td>bdcdd120-9979-491e-8683-d9a4550bdd</td>
</tr>
<tr>
<td>Application ID</td>
<td>Client ID</td>
<td>e6821ea9-e5c5-428a-b50c-7ea62993b2d</td>
</tr>
<tr>
<td>Application key</td>
<td>Secret key</td>
<td>d348050-f893ccb1</td>
</tr>
<tr>
<td>Subscription ID</td>
<td>Account ID</td>
<td>4803be48c3ee89-96d2-4fed-bedb-22e49013c16</td>
</tr>
</tbody>
</table>

2. Log in to the Azure portal and click Azure Active Directory.
3. In the Azure Active Directory blade, select Properties.
4. On the Properties blade, navigate to the Directory ID and then click the Copy to Clipboard ( ) button.
5. Paste the **Directory ID** value (the UUID) into the text file.  
   To complete the next step, you must have permission to create and register an application. If not, ask your Active Directory admin for help.

6. On the **Azure Active Directory** blade, click **App registrations** blade, click **+ Add** or **+ New application registration**, enter the following values, and then click **Create**.
Create

* Name

ServiceNow Integration

Application type

Web app / API

* Sign-on URL

http://www.myinstance.com
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name for the application and its integration credentials. For example, ServiceNow Integration.</td>
</tr>
<tr>
<td>Application type</td>
<td>Select Web app / API.</td>
</tr>
<tr>
<td>Sign-on URL</td>
<td>URL that will access Azure data — the URL of the ServiceNow instance.</td>
</tr>
</tbody>
</table>

7. In the list, click the **DISPLAY NAME** of the new application.

![Image of Application Table]

- a) Point to the **Display Name** and then click the **Copy to Clipboard** button. Paste the **Display Name** into the text file.

- b) Point to the **Application ID** and then click the **Copy to Clipboard** button. Paste the **Application ID** into the text file.
8. In the navigation pane, under **API ACCESS**, click **Keys**.

9. Specify the following values, and then click **Save**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Unique description of the key.</td>
</tr>
<tr>
<td>Expires</td>
<td>Your organization may apply policies to restrict key durability. Select the appropriate duration.</td>
</tr>
</tbody>
</table>
The key is saved and the key value appears in the **VALUE** column.
10. Copy the key **VALUE** and paste the value into the text file as the **Application key** value.

**Note:** Do not leave the page until you have copy/pasted the key value. The value will not appear again.

11. Enable the service principal to assign policy to a resource group.
By default, an Azure service principal has the Contributor role. The Contributor role cannot assign policy to a resource group. Create a custom role using the Azure command-line interface, as follows:

a) A user with the Owner role or the User access admin role should run the following command. In the example, you create the Policy Assignment Operator role:

```
az role definition create --role-definition ' {
  "Name": "Policy Assignment Operator",
  "Description": "Lets you assign policy",
  "Actions": ["Microsoft.Authorization/policyAssignments/write",
               "Microsoft.Authorization/policyAssignments/delete",
               "Microsoft.Authorization/policyAssignments/read"],
  "AssignableScopes": ["/subscriptions/<Your-subscription-ID-here>"]
}'
```

b) Assign the Policy Assignment Operator role to the service principal.

If you do not have access to the command-line interface or cannot create a custom role, then another option to enable the service principal to assign policy is to assign the User access administrator role to the service principal. This option, however, is risky because the role has other powerful access.

12. To enable the service principal to work with multiple Azure subscriptions, you must perform the following procedure for each subscription:

a) In the Azure portal, navigate to Subscriptions.

b) Copy the Subscription ID and paste it into the text file.

c) Open the subscription, select Access Control (IAM) from the menu, and then click + Add. Specify the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value to set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role</td>
<td>Contributor</td>
</tr>
<tr>
<td>Assign access to</td>
<td>Azure AD user, group, or application</td>
</tr>
</tbody>
</table>
Create a record of the service principal credentials on the ServiceNow instance so that Cloud Management processes can access Azure data. See Store the Azure service principal credentials in the instance.

**Store the Azure service principal credentials in the instance**

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you first create Azure service principal credentials in the Azure Portal. You then securely store the credentials in a service account in your instance.

Role required:
- Azure Portal administrator
- Cloud Management: admin or sn_cmp.cloud_admin

1. Create an Azure service principal and open the text file that you created during the procedure.
2. In the Cloud Admin Portal, navigate to Manage > Credentials.
3. Click New and then select Azure Service Principal.
4. Specify the following values on the Azure Service Principal form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the service principal to register with the instance. For example, Azure service principal credentials.</td>
</tr>
<tr>
<td>Authentication Method</td>
<td>Select <strong>Client Secret</strong>. The <strong>Secret key</strong> field appears when you select Client Secret.</td>
</tr>
</tbody>
</table>

5. Copy/paste values from the Azure-Credentials.txt text file into the remaining fields.
6. Select the appropriate **EA credential** from the list, select the **Active** check box, and then click **Save** to create the record.

7. Click the **Discover Subscriptions** related link to find all subscriptions that are associated with the Azure service principal.

   The instance creates a service account for each discovered subscription.

8. The **Azure Subscriptions** related list displays all subscriptions that are associated with the Azure service principal. Click a subscription to view the service account that was created for the subscription.

9. Click a discovery status entry in the **Credential Discovery Status** list to view the associated discovery log. Each time that you click **Discover Subscriptions**, the instance generates a new discovery status and lists the status in the **Credential Discovery Status** list.

### Add an Azure service account

During Cloud Management Day 1 setup, you added one service account to the cloud account. To compartmentalize your infrastructure or to include different datacenters, you can add another service account. A particular datacenter, however, cannot be selected in more than one service account in a cloud account.

**Role required:** sn_cmp.cloud_admin

1. Open the **Azure-Credentials.txt** text file that you created during the **Create an Azure service principal** procedure.

2. Navigate to **Cloud Management > Service Accounts**.

3. Click **New**, enter a unique and meaningful **Name**, and then fill in the form.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account ID</td>
<td>Paste the Azure Subscription ID value that you copied from the Azure Portal into the text file.</td>
</tr>
<tr>
<td>Discovery credentials</td>
<td>Select the name of the credentials that you created in the Store the Azure service principal credentials in the instance procedure. In the example, you used the name Azure service principal credentials.</td>
</tr>
<tr>
<td>Datacenter URL</td>
<td>Leave this field blank.</td>
</tr>
<tr>
<td>Datacenter type</td>
<td>Select Azure Datacenter.</td>
</tr>
<tr>
<td>Datacenter discovery status</td>
<td>Auto-generated value: Status and timestamp of the last execution of Discovery on the datacenter.</td>
</tr>
<tr>
<td>Should pull events</td>
<td>Enables Cloud Management to collect events from resource types in Azure LDCs in the sn_cmp_cloud_event_list.</td>
</tr>
</tbody>
</table>

4. Click **Update** or **Submit**.

The system creates the service account and displays the list of all discovered datacenters.

After you complete all procedures in this initial "Day 1" setup, you can create additional cloud accounts and service accounts to organize and compartmentalize your cloud infrastructure.
Set up a cloud account for Azure

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

Role required: sn_cmp.cloud_admin

See Cloud accounts for more information about cloud accounts.

**Note:** Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In Azure, the region-specific containers for virtual resources are called regions. In Cloud Management, regions are called datacenters or logical datacenters (LDCs). The term "logical" is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.

1. In the Cloud Admin Portal, navigate to Manage > Cloud Accounts.
2. Click New, enter a unique and meaningful Name and Description, and then select the cloud Provider for the account (Azure).
3. Click Next.

Based on the provider you select, the **Service Account** field is populated and a list of discovered datacenters appears. If no datacenters appear, click **Discover Datacenters** to find them. See Discover all datacenters in a service account on-demand for details.
4. When the datacenters appear, select one or more datacenters to include in the cloud account and then click Save. The Cloud Account dashboard appears. The datacenters that you selected appear on the Datacenters tab.

5. Click a datacenter. The following lists appear:
   - **Resources**: Cloud resources for the current datacenter. Run Discovery to populate the CMDB for the datacenter and populate the tab. See Discover all datacenters in a service account on-demand.
   - **Capacity Limits**: Limits on virtual CPUs, virtual networks, storage volume size, and other services. See Set capacity limits on user requests for resources.
   - **Discovery Log**: The process creates a log record for each action associated with a discovery status.

6. Repeat the process to add as many service accounts as needed.

**Configure the Azure Alert service to auto-update the CMDB**

The Azure Alert service can auto-update CI data in the CMDB whenever Cloud Management or your Azure account makes a life-cycle state or configuration change to an Azure resource. As a result, the CI data in the CMDB is updated without having to wait for Discovery to run.

- Role required: sn_cmp.cloud_event_integration
- The following Cloud Management plugins must be activated:
  - Cloud Management (com.snc.cloud.mgmt)
  - Cloud Management Core (com.snc.cloud.core)
  - Cloud API (com.snc.cloud.api)
  - Cloud Config Management (com.snc.config.mgmt)
- Discovery has successfully finished running with at least one resource group
- A valid MID Server with either ALL or 'Cloud Management' and 'Azure' capabilities must always be available.
- A cloud account with Azure subscriptions (service accounts) and associated logical datacenters are discovered.

**Note:** The alert configuration continues to ping alerts even when the MID Server is down. This causes a high number of errors in the ECC queue and you need to manually deactivate the Azure Alert configuration if there is no MID Server available.

When an alert sends an update, the instance processes the event and creates or updates the CI entry in the CMDB and the CI information in the User Portal. Each event is saved as a record in the Cloud Events sn_cmp_cloud_event table. Cloud Management supports all the alerts supported in a Resource Group if an alert/event has been generated to the resources in a Resource Group. You should configure only one rule for a service account. Alert rules are created for all resources within the service account.

2. On the Azure tab, click New and fill in the form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Account</td>
<td>Service account details which has the datacenters and Resource Groups discovered successfully.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Resource Group</td>
<td>Resource group which stores the alert rules. Select the service account that you created for an Azure account, as described in <a href="#">Add an Azure service account</a>. Cloud Management creates alert rules for all resources in the service account.</td>
</tr>
<tr>
<td>User/Password</td>
<td>Azure does not currently support authentication. To receive Azure Alert events, however, you must provide a user name for an account with the sn_cmp.cloud_event_integration role. In addition, you must disable Authentication for the Cloud Event Scripted REST API as follows:</td>
</tr>
<tr>
<td></td>
<td>1. As System Admin, open the Scripted REST API module.</td>
</tr>
<tr>
<td></td>
<td>2. Open the record for Cloud Events.</td>
</tr>
<tr>
<td></td>
<td>3. Under Resources, open <a href="#">Cloud Config Event Post</a>.</td>
</tr>
</tbody>
</table>
| Status (read only)    | - Updating  
- Activated  
- Deleting  
- Deleted                                                                                                                                                                                                   |

3. Click **Submit**.  
   On successfully configuring the Azure alert service, the status changes from **Started** to **Active**.  

4. Run discovery on the service account:  
   a) Navigate to **Cloud Management > Service Accounts**.  
   b) Click the name of a service account.  
   c) Click **Create Discovery Schedule**.  
      The **Create Discovery Schedule** related link does not appear unless you discover the associated datacenters. Click **Discover datacenters** first, and then click **Create Discovery Schedule**.  
   d) Configure the schedule. See [Schedule discovery of all resources in a service account](#) for details on all the fields.  
      Keep the **Discover** field **Cloud Resources**.  
   e) Click **Discover now** or wait for the scheduled discovery to run. 

After you complete these steps, events are saved in the Cloud Event (sn_cmp_cloud_event) table, and the corresponding CIs will be created, updated, or deleted. The horizontal discovery process does not run, unless the event is database-related or related to Azure web sites, so there are no Discovery Status or ECC Queue records.  
- All Azure events are populated with Source: **azure activity log**.
The **Event Name** field displays the actual event generated from the Azure console.

- Events are generated with the associated in the instance. Cloud Management identifies the VMs based on the **Resource ID**, not on the resource name.
- The **Resource Type** field indicates the resource type which generates the event.

Once an event has been generated, the Cloud Event Scheduler executes scheduled jobs in the background.

- By default, the Cloud Event Scheduler executes scheduled jobs every 10 seconds. You can modify the default repeat interval based on your requirements.
- The scheduled job passes the event payload to the Identification and Reconciliation Engine (IRE). The IRE then takes the necessary actions to modify the state of the resources based on the event.

For events on Azure databases and on Azure web servers, the horizontal discovery process triggers the Azure Database pattern and then a **Discovery Status record** and **ECC Queue records** are created. You can use these to track the status of the event processing just as you would track the progress of a standard CI discovery.

### Define the schedule for downloading Azure billing data

Define the scheduled job that regularly uses a MID Server to download billing data from the provider. Cloud Management saves the data in a cost table and uses the information to generate reports.

**Role required:** sn_cmp.cloud_governor

You must have an API Access Key credential for a Microsoft Enterprise Agreement EA for all Azure accounts that you want billing information for.

This task involves two procedures:

1. Fill in the Azure Enterprise Agreement Credentials form with information that you copy from the Azure Portal. This procedure enables the instance to use the EA credentials to download Azure billing data.

2. Specify the schedule for downloading Azure billing information.

   1. In the Cloud Admin Portal, navigate to **Manage > Credentials**.
   2. Click **New** and then select **Azure Enterprise Agreement Credentials**.
   3. Enter a unique and descriptive **Name** for the credential and then fill in the form.
<table>
<thead>
<tr>
<th>Enrollment number</th>
<th>Follow this procedure to obtain the <strong>Enrollment number</strong>:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>1.</strong> Log in to ea.azure.com</td>
</tr>
<tr>
<td></td>
<td><strong>2.</strong> The enrollment number appears under the Windows icon. In the example, the enrollment number is ABC01234.</td>
</tr>
<tr>
<td></td>
<td><strong>3.</strong> Copy/paste the value into the <strong>Enrollment number</strong> field on the form.</td>
</tr>
</tbody>
</table>
Follow this procedure to obtain the **Access key**:

1. Log in to ea.azure.com
2. In the **Reports** section, click the **Download Usage** tab and then click **API Access Key**.

   ![Image of API Access Key](image)

   **Note:** For a new account, you may need to generate a key.

3. This account has a primary key. Click **expand key** to display the key in the text box.

   ![Image of expanded key](image)
4. Click **Submit**.
   You created the Azure EA credential. Next, you specify the scheduled job that downloads Azure billing information.

5. In the Cloud Admin Portal, navigate to **Analyze > Billing**.

6. On the **Billing Schedules** tab, click **New**, enter a unique and meaningful **Name**, and then fill in the form.

<table>
<thead>
<tr>
<th>Provider</th>
<th>Cloud provider that the service account belongs to. Select <strong>Azure</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential</td>
<td>Select the Azure EA credential that you created earlier.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to activate the download job.</td>
</tr>
<tr>
<td>Run</td>
<td>Frequency that the download job should run.</td>
</tr>
<tr>
<td>Time</td>
<td>Time of day that the download job should run.</td>
</tr>
</tbody>
</table>

7. Click **Save**.
   Typically, you must wait to request data for about 48 hours after you set up the connection to a provider. To run the schedule on demand, click **Execute Now**. On the **Retrieve Billing Data** window that appears, select a date range and then click **Download**.

*View cloud costs on the Billing dashboard.*

*Optional: Configure the download size of Azure billing data*

A single set of cloud billing data can be large. The MID Server, therefore, sends the data to the ECC queue in manageable chunks. You can configure a system property to limit the size of each chunk to avoid performance issues caused by large data transfers.

**Role required: admin**

1. Open the System properties table by entering `sys_properties.list` in the application filter.
2. Filter the list to open the `sn_cmp.billing.page_size` property.
   The default value is 3 MB.
3. Modify the value as needed. The maximum is 5 MB.
4. Save the property.

---

**Day 1 setup guide for Cloud Optimization: Google Cloud Connector**

To set up the ServiceNow® Cloud Optimization: Google Cloud Connector application for the very first time, you perform the procedures in this “Day 1” setup guide. Be sure to perform the procedures in order. After you have performed Day 1 setup, you can perform optional Day 2 setup and configuration procedures as needed and in any order. Detailed instructions for each procedure follow this overview.

**Roles required to set up Google Cloud Connector**

- Operations in the Google Cloud Console require the Google administrator role.
- Operations in Cloud Management require the `sn_cmp.cloud_admin` role.
Domain Separation

The Google Cloud Connector app does not support domain separation.

About terms that Cloud Management uses

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In the Google Cloud Platform, the region-specific containers for virtual resources are called regions. In Cloud Management, regions are called datacenters or logical datacenters (LDCs). The term "logical" is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.

Quick overview of the setup process

Your Day 1 setup process includes these tasks:

- If needed: Request the Cloud Management application.
- Get the Connector app and supporting apps on the ServiceNow Store and set them up.
- Assign appropriate roles to cloud admins and end users.
- On the provider portal, collect your account settings and the credentials that the Discovery process will use (through a MID Server) to programatically access your provider accounts. Securely associate the account settings and credentials with a service account in Cloud Management.
- Set up the MID Servers that will handle secure communications with the provider API endpoints.
- Set up a cloud account to represent your entire managed cloud infrastructure and set up a service account that works with one of your provider accounts. You specify which datacenters in the service account should be included in the cloud account. (Later, on "Day 2", you can set up additional cloud accounts and service accounts from the same or other providers.)
- To populate the CMDB with resource data for all datacenters, you manually run the Discovery process on each datacenter in the service account. Then, to ensure that the data continues to be updated, you configure Discovery to run on a regular schedule. Your cloud account might look like this:
• Providers offer services that can auto-update the CMDB whenever a create/modify/terminate life-cycle change or configuration update occurs to a resource. You can configure the service to integrate directly with Cloud Management.

What you will do to integrate your Google Cloud Platform cloud accounts

Detailed instructions for each procedure follow this overview.

1. Get the Google Cloud Connector app on the ServiceNow Store

Google Cloud Connector works with Cloud Management. After you have set up Cloud Management, visit the ServiceNow Store website to get Google Cloud Connector and supporting plugins and apps.

2. Assign roles to Google Cloud Platform users

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

3. Set up MID Servers to connect Google Cloud Connector to your Google Cloud Platform account

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

Note: Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.

4. Specify the credentials that Google Cloud Connector uses to access Google Cloud Platform data

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you open the Google Cloud Console to identify the Google Cloud Platform project that will have programmatic access to your Google Cloud Platform data. You then securely store the credentials in a service account in your instance.

5. Set up a cloud account and service account for Google Cloud Platform

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter.

6. Configure the Google Stackdriver Logging service to auto-update the CMDB

You can activate the Google Stackdriver Logging service to auto-update CMDB CI data whenever Google Cloud Connector or your Google account makes a life-cycle state or configuration change to a Google Cloud Platform resource. As a result, the CI data in the CMDB is updated without having to wait for Discovery to run.
Next steps

When you have finished all Day-1 and Day-2 procedures in this setup guide, see the *Cloud Management administration guide* for information on using the Cloud Management application in your organization.

Assign roles to Google Cloud Platform users

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

Role required: user_admin or admin

Assign the following roles to groups and users as appropriate:

<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root administrator</td>
<td>Highest level of application access for Cloud Management.</td>
<td>All</td>
</tr>
<tr>
<td>[sn_cmp.cmp_root_admin]</td>
<td>Configures the Cloud Management application and sets up the cloud infrastructure.</td>
<td>Cloud infrastructure: Service accounts and cloud account Networks and IPAMp Settings for provider services that auto-update the CMDB: • AWS Config • Azure Alert • Google Stackdriver Logging • IBM Cloud Update • VMware Events</td>
</tr>
<tr>
<td>Cloud administrator</td>
<td>Monitors overall cloud usage and enforces compliance of the organization’s rules, quotas, and policies. Also manages tags and permissions to various objects.</td>
<td>Governance: • Policies • Pools • Quotas • Permissions</td>
</tr>
<tr>
<td>[sn_cmp.cloud_admin]</td>
<td>Creates blueprints, ARM and CloudFormation templates, and catalog items.</td>
<td>Cloud Service Design: • Cloud templates • Blueprints • Blueprint catalog items • Resource blocks</td>
</tr>
<tr>
<td>Governor</td>
<td>Requests and manages stacks and resources.</td>
<td>Cloud User Portal When you assign the role to a group, all members of the group share quota limitations and ownership of certain resources.</td>
</tr>
<tr>
<td>[sn_cmp.cloud_governor]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[sn_cmp.cloud_service_user]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
</table>
| Cloud operator [sn_cmp.cloud_operator] | Monitors and troubleshoots the Cloud Management application. | Dashboards and reports:  
- Cloud Operations Dashboard  
- Cloud Root Cause Analysis  
- Cloud Orchestration Trail  
- Cloud API Trail |
| Note: Not supported by IBM Cloud Connector. |
| Cloud Event Integration [sn_cmp.cloud_event_integration] | Authorizes access to the instance for external services that auto-update the CMDB when cloud events occur. This role gives the access to the cloud event REST endpoint. | Cloud infrastructure:  
- Service accounts and cloud account  
- Networks and IPAM  
Settings for provider services that auto-update the CMDB:  
- AWS Config  
- Azure Alert  
- Google Stackdriver Logging  
- VMware Events |
| Cloud Group administrator [sn_cmp.cloud_group_admin] | Grants admin access to any group that you belong to. | |

---

**Set up MID Servers to connect Google Cloud Connector to your Google Cloud Platform account**

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

Role required: admin or sn_cmp.cloud_admin

- Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.
- You can configure the MID Server for Cloud Management to use a proxy server. See **Using a proxy server for your MID Server**.

1. You configure the connection to a MID Server in the same way as for any other applications that use MID Servers, like Discovery and Service Mapping. See **MID Server** for instructions on setting up and configuring MID Servers.

   **Note:** Do not configure the MID Server Application related list.

2. Add the **Google** and **Terraform** capabilities to the MID Server. See **Configure MID Server capabilities** for instructions. Alternatively, you can add the **ALL** capability, which includes the **Google** and **Terraform** capabilities.
Using a proxy server for your MID Server

You can configure the MID Server for Cloud Management to use a proxy server. Using a proxy server supports all cloud-based activities such as running Discovery, billing downloads, provisioning virtual machines, and running life-cycle operations on virtual machines.

Proxy server limitations

- Windows or Linux platforms only are supported.
- The Google Cloud Console integration does not support downloading billing data.
- VMware is not supported.
- Remote PowerShell scripts cannot be executed.
- Custom APIs may not work.

Supported proxy server configurations for Cloud Management

<table>
<thead>
<tr>
<th>Authentication/No Authentication</th>
<th>Authentication type</th>
<th>Proxy Server type</th>
</tr>
</thead>
<tbody>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>NTLM</td>
<td>HTTP/HTTPS</td>
</tr>
</tbody>
</table>

Qualified proxy server configurations for Cloud Management

Qualified configuration settings

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Operating system</th>
<th>Proxy server</th>
<th>Authentication mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration 1</td>
<td>Linux</td>
<td>None</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Configuration 2</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>None</td>
</tr>
<tr>
<td>Configuration 3</td>
<td>Linux</td>
<td>Squid (HTTPS)</td>
<td>Local</td>
</tr>
<tr>
<td>Configuration 4</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>Active Directory</td>
</tr>
</tbody>
</table>

Qualified configurations

<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS, Azure, and IBM Cloud</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Billing download</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Google Cloud Platform</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>VMware</td>
<td>Not Applicable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Specify the credentials that Google Cloud Connector uses to access Google Cloud Platform data

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you open the Google Cloud Console to identify the Google Cloud Platform project that will have programmatic access to your Google Cloud Platform data. You then securely store the credentials in a service account in your instance.

Roles required:
- Operations in the Google Cloud Console require the Google administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings.

1. If you have already generated the credentials that Google Cloud Connector should use, skip to the final step. Otherwise, continue with the next step.
2. On the Google Cloud Console dashboard, select the project.
3. Navigate to APIs & Services > Credentials.
4. On the Credentials tab, click Create credentials > Service account key and then create the key. Specify the JSON key type.
5. Identify the credentials that Google Cloud Connector and Discovery on your instance should use to access Google Cloud Platform data:
   a) Open the JSON key file in a text editor.
b) In the Cloud Admin Portal, navigate to Manage > Credentials, click New and then select Google Credentials.

c) Specify the Name of the credentials to register with the instance. For example, enter Google credentials-1 and then copy/paste values from the JSON key file into the form fields. Paste the private_key value into the Secret key field and the client_email value into the Email field.

d) Select the Active check box to use the credentials and then click Submit.

Set up a cloud account and service account for Google Cloud Platform

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter. A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

- Operations in the Google Cloud Console require the Google administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.

When you finish the Day-1 setup process, your cloud account might look like this:

2. Select or create the cloud account:
   - Select an existing cloud account:
     1. On the Cloud Account page, click Configure.
     2. On the General Information page, click Next and then continue with the next step.
   - Create a cloud account:
     1. On the General Information page, click New and then enter a unique and meaningful Name and Description for the cloud account.
     2. Select the provider.
     3. Click Next and then continue with the next step.

3. On the Datacenters page, create a Service Account: Click the + next to the Service Account field, enter a unique and meaningful Name for the service account and then fill in the form.

4. From the JSON key file that is associated with the service account, copy/paste the project_id value into the Account ID field.
5. In the **Discovery credentials** field, select the appropriate credentials for the service account. These are the credentials that you generated in the Specify the credentials that Google Cloud Connector uses to access Google Cloud Platform data procedure.

6. Leave the **Datacenter URL** field blank.

7. Fill in the remaining fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datacenter type</td>
<td>Select <strong>Google Cloud Platform Datacenter</strong>.</td>
</tr>
<tr>
<td>Datacenter discovery status</td>
<td>Select <strong>Google Cloud Platform Datacenter</strong>.</td>
</tr>
<tr>
<td>Should pull events</td>
<td>Enables Cloud Management to collect events from resource types in Google Project LDCs in the sn_cmp_cloud_event_list.</td>
</tr>
</tbody>
</table>

8. Click **Submit** to create the service account.

9. Click **Discover Datacenters**.

10. When the datacenters appear, select one or more datacenters to include in the cloud account and then click **Save**.

11. Click a datacenter.

The following lists appear:

- **Resources**: Cloud resources for the current datacenter. Run Discovery to populate the CMDB for the datacenter and populate the tab. See *Discover all datacenters in a service account on-demand*.
- **Capacity Limits**: Limits on virtual CPUs, virtual networks, storage volume size, and other services. See *Set capacity limits on user requests for resources*.
- **Discovery Log**: The process creates a log record for each action associated with a discovery status.

12. Repeat the process to add as many service accounts as needed.

After you complete all procedures in this initial “Day 1” setup, you can create additional cloud accounts and service accounts to organize and compartmentalize your cloud infrastructure.
Configure the Google Stackdriver Logging service to auto-update the CMDB

You can activate the Google Stackdriver Logging service to auto-update CMDB CI data whenever Google Cloud Connector or your Google account makes a life-cycle state or configuration change to a Google Cloud Platform resource. As a result, the CI data in the CMDB is updated without having to wait for Discovery to run.

- Google Cloud Platform administrator
- Role required: for Cloud Management: sn_cmp.cloud_event_integration
- The following Cloud Management plugins must be activated:
  - Cloud Management (com.snc.cloud.mgmt)
  - Cloud Management Core (com.snc.cloud.core)
  - Cloud API (com.snc.cloud.api)
  - Cloud Config Management (com.snc.config.mgmt)

- Activate Discovery and Service-mapping patterns.
- Discovery has successfully discovered logical datacenters of the Google project. For more information see, Set up a cloud account and service account for Google Cloud Platform

Note: Ensure that you select the Should pull events check box in the Google Service Account page, before you execute datacenter discovery.

Cloud Management supports StackDriver Monitoring API alerts from a given project in Google Cloud Platform. When an alert/event has been generated to the resources in a project, Cloud Management captures.

Create and manage alerting policies with the Stackdriver Monitoring console, the Stackdriver Monitoring API, and Cloud SDK. Each policy specifies the following:

- Conditions that identify that a resource or a group of resources are in an unhealthy state.
- Notifications sent through email, SMS, or other channels to inform that a resource state is unhealthy.
- Documentation or information that can be included in some types of notifications to help your support team resolve the issue.

When events trigger conditions in one of your alerting policies, Stackdriver Monitoring creates and displays an incident in the Stackdriver Monitoring console.

1. Log in to your Google Cloud Platform account. For your project, navigate to APIs & Services > Library.
2. Enable both the Stackdriver API and Stackdriver Logging API.
   a) In the library, search for Stackdriver.
   b) Click the Stackdriver API card.
   c) On the Stackdriver API page, click Enable.
d) Repeat for the Stackdriver Logging API.

3. By default, a scheduled job on your instance (Google Cloud Event Scheduler) polls the Google Stackdriver Logging service for updates every 5 minutes. To modify the schedule: On your instance, navigate to **System Scheduler > Scheduled Jobs** and then click **Google Cloud Event Scheduler** and then update the **Repeat** setting as needed.
4. **Click Update.**
   - Once the Google Cloud Platform generates an event and Cloud Management receives the event information, the ‘GCP-Events-job’ scheduled jobs execute in the background.
   - The scheduled job passes the event payload to the Identification and Reconciliation Engine (IRE). The IRE then takes the necessary actions to modify the state of the resources based on the event.

**(Optional) Add a Google Cloud Platform service account to the cloud account**

During Cloud Management Day 1 setup, you added one service account to the cloud account. To compartmentalize your infrastructure or to include different datacenters, you can add another service account. A particular datacenter, however, cannot be selected in more than one service account in a cloud account.

*Role required: sn_cmp.cloud_admin*

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter.

In this example, you added the service account named **ProviderB-ServiceAccount-1** and selected three datacenters to include in the cloud account:
1. Navigate to **Cloud Management > Service Accounts**.
2. Click **New**, enter a unique and meaningful **Name**, and then fill in the form.
3. Fill in the remaining fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datacenter type</td>
<td>Select <strong>Google Cloud Platform Datacenter</strong>.</td>
</tr>
<tr>
<td>Datacenter discovery status</td>
<td>Auto-generated value: Status and timestamp of the last execution of Discovery on the datacenter.</td>
</tr>
</tbody>
</table>

4. Click **Update** or **Submit**.
   The system creates the service account and displays the list of all discovered datacenters.
5. Repeat the process to add as many service accounts as needed.

---

**Note:** In a cloud account, you cannot select a particular datacenter in two different service accounts.
Day 1 setup guide for Cloud Optimization: IBM Cloud Connector

To set up the ServiceNow® Cloud Optimization: IBM Cloud Connector application for the very first time, you perform the procedures in this “Day 1” setup guide. Be sure to perform the procedures in order. After you have performed Day 1 setup, you can perform optional Day 2 setup and configuration procedures as needed and in any order. Detailed instructions for each procedure follow this overview.

Roles required to set up IBM Cloud Connector

- Operations in the IBM Cloud Console require the IBM administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.

Domain Separation

The IBM Cloud Connector app does not support domain separation.

Known issue

In some cases, Discovery for IBM Cloud is not able to change the state of some CIs that have been decommissioned in the cloud. In such a case, when an end user launches a cloud catalog item, the CI may still appear as a user-selectable value on the form (because the state hasn’t been updated) and may result in a failed request. Workaround: The cloud administrator can change the state of the affected CI to an appropriate value to remove the CI from the cloud catalog form.

About terms that Cloud Management uses

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In IBM Cloud, the region-specific containers for virtual resources are called locations. In Cloud Management, locations are called datacenters or logical datacenters (LDCs). The term "logical" is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.

What you will do to integrate your IBM Cloud Connector accounts

Detailed instructions for each procedure follow this overview.

1. Get the IBM Cloud Connector app on the ServiceNow Store

IBM Cloud Connector works with Cloud Management. After you have set up Cloud Management, visit the ServiceNow Store website to get IBM Cloud Connector and supporting plugins and apps. For details on system requirements, view the application listing on the ServiceNow Store website.

2. Assign roles to IBM Cloud Connector users

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.
3. Set up MID Servers to support IBM Cloud Connector

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

**Note:** Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.

3. Create a Terraform config provider and run Discovery

A config provider is a secure record on your instance that stores the credential and access information for a particular configuration management server (for example, a host running Ansible, or Terraform). Discovery uses the information in the config provider record to access the server to get information on each resource template in the repository. You use the templates to generate catalog items that cloud users can order from the Cloud User Portal.

4. Create an IBM Cloud service account

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter.

5. Discover all datacenters in a service account on-demand

Discovery uses the information in the service account to identify all logical datacenters associated with the provider account.

6. Set capacity limits on user requests for resources

Capacity limits place restrictions on the attributes of cloud resources such as the number of virtual machines, virtual CPUs, or aggregate storage. You can set limits on resources separately for each logical datacenter in a cloud account.

**Next steps**

When you have finished all Day-1 and Day-2 procedures in this setup guide, see the Cloud Management administration guide for information on using the Cloud Management application in your organization.

**Assign roles to IBM Cloud Connector users**

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities. Many of the roles might already be assigned for Cloud Management.

Role required: user_admin or admin

Assign the following roles to groups and users as appropriate:

<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root administrator [sn_cmp.cmp_root_admin]</td>
<td>Highest level of application access for Cloud Management.</td>
<td>All</td>
</tr>
<tr>
<td>Descriptive name and role name</td>
<td>Description and tasks</td>
<td>Access rights in Cloud Management</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td><strong>Cloud administrator</strong></td>
<td>Configures the Cloud Management application and sets up the cloud infrastructure.</td>
<td>Cloud infrastructure: Service accounts and cloud account Networks and IPAMp Settings for provider services that auto-update the CMDB:  * AWS Config  * Azure Alert  * Google Stackdriver  Logging  * IBM Cloud Update  * VMware Events</td>
</tr>
<tr>
<td>[sn_cmp.cloud_admin]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Governor</strong></td>
<td>Monitors overall cloud usage and enforces compliance of the organization’s rules, quotas, and policies. Also manages tags and permissions to various objects.</td>
<td>Governance:  * Policies  * Pools  * Quotas  * Permissions</td>
</tr>
<tr>
<td>[sn_cmp.cloud_governor]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Service Designer</strong></td>
<td>Creates blueprints, ARM and CloudFormation templates, and catalog items.</td>
<td>Cloud Service Design:  * Cloud templates  * Blueprints  * Blueprint catalog items  * Resource blocks</td>
</tr>
<tr>
<td>[sn_cmp.cloud_service_designer]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cloud user</strong></td>
<td>Requests and manages stacks and resources.</td>
<td>Cloud User Portal  When you assign the role to a group, all members of the group share quota limitations and ownership of certain resources.</td>
</tr>
<tr>
<td>[sn_cmp.cloud_service_user]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cloud operator</strong></td>
<td>Monitors and troubleshoots the Cloud Management application.</td>
<td>Dashboards and reports:  * Cloud Operations Dashboard  * Cloud Root Cause Analysis  * Cloud Orchestration Trail  * Cloud API Trail</td>
</tr>
<tr>
<td>[sn_cmp.cloud_operator]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Descriptive name and role name</td>
<td>Description and tasks</td>
<td>Access rights in Cloud Management</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Cloud Event Integration</td>
<td>Authorizes access to the instance for external services that auto-update the CMDB when cloud events occur. This role gives the access to the cloud event REST endpoint.</td>
<td>Cloud infrastructure: • Service accounts and cloud account • Networks and IPAM</td>
</tr>
<tr>
<td>[sn_cmp.cloud_event_integration]</td>
<td></td>
<td>Settings for provider services that auto-update the CMDB: • AWS Config • Azure Alert • Google Stackdriver Logging • VMware Events</td>
</tr>
</tbody>
</table>

**Note:** Not supported by IBM Cloud Connector.

| Cloud Group administrator                  | Grants admin access to any group that you belong to.                                                                                                                                                                |                                                                                                              |
| [sn_cmp.cloud_group_admin]                |                                                                                                                                                                                                                      |                                                                                                              |

---

**Set up MID Servers to support IBM Cloud Connector**

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

Role required: admin or sn_cmp.cloud_admin

- Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.
- You can configure the MID Server for IBM Cloud Connector to use a proxy server. See [Using a proxy server for your MID Server](#).

1. If you have already configured a MID Server to support Cloud Management, then skip this step. You configure the connection to a MID Server in the same way as for any other applications that use MID Servers, like Discovery and Service Mapping. See [MID Server](#) for instructions on setting up and configuring MID Servers.

   **Note:** Do not configure the MID Server Application related list.

2. Add the **IBM** and **Terraform** capabilities to the MID Server. See [Configure MID Server capabilities](#) for instructions. Alternatively, you can add the **ALL** capability, which includes the **IBM** and **Terraform** capabilities.

   **Using a proxy server for your MID Server**

   You can configure the MID Server for Cloud Management to use a proxy server. Using a proxy server supports all cloud-based activities such as running Discovery, billing downloads, provisioning virtual machines, and running life-cycle operations on virtual machines.

   **Proxy server limitations**

   - Windows or Linux platforms only are supported.
- The Google Cloud Console integration does not support downloading billing data.
- VMware is not supported.
- Remote PowerShell scripts cannot be executed.
- Custom APIs may not work.

**Supported proxy server configurations for Cloud Management**

<table>
<thead>
<tr>
<th>Authentication/No Authentication</th>
<th>Authentication type</th>
<th>Proxy Server type</th>
</tr>
</thead>
<tbody>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>NTLM</td>
<td>HTTP/HTTPS</td>
</tr>
</tbody>
</table>

**Qualified proxy server configurations for Cloud Management**

**Qualified configuration settings**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Operating system</th>
<th>Proxy server</th>
<th>Authentication mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration 1</td>
<td>Linux</td>
<td>None</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Configuration 2</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>None</td>
</tr>
<tr>
<td>Configuration 3</td>
<td>Linux</td>
<td>Squid (HTTPS)</td>
<td>Local</td>
</tr>
<tr>
<td>Configuration 4</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>Active Directory</td>
</tr>
</tbody>
</table>

**Qualified configurations**

<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS, Azure, and IBM Cloud</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Google Cloud Platform</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Cloud provider</td>
<td>Functionality</td>
<td>Configuration 1</td>
<td>Configuration 2</td>
<td>Configuration 3</td>
<td>Configuration 4</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>-----------------</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>VMware</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

Create an IBM Cloud service account

During Cloud Management Day 1 setup, you added one service account to the cloud account. To compartmentalize your infrastructure or to include different datacenters, you can add another service account. A particular datacenter, however, cannot be selected in more than one service account in a cloud account.

Role required: sn_cmp.cloud_admin

A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter.

In this example, you added the service account named **ProviderB-ServiceAccount-1** and selected three datacenters to include in the cloud account:
Note: Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings.

1. On your instance, navigate to **Orchestration > Credentials & Connections > Credentials**.
2. Click **New**, select **IBM Credentials**, and then enter a unique and meaningful **Name** for the credential set (in this example, **IBM Van Credentials**). If you use more than one set of IBM credentials, set the **Order** value (credential sets with lower values are checked first).
   In the following steps, you paste the IBM credential values into the Credentials form on your instance.
3. Log in to your Softlayer account at control.softlayer.com.
4. Navigate to **Account > Users > User List**.
5. On the Users page, click **View** for the appropriate **API KEY** (or generate an API key if needed).
6. From the API Key popup:
   a) Paste the full API User value into the Softlayer API User field on your instance.
   b) Paste the API Key value into the Softlayer API Key field.

7. Log in to your Bluemix account at console.bluemix.net.
9. Click Platform API Keys, click the Create button, and then enter a Name and Description for the key.
10. On the success page, click **Download** to download the key to a local JSON file.

11. Open the file (typically named `apiKey-n.json`), paste the **apiKey** value into the **Bluemix API Key** field on your instance, and then click **Update**.
You have created a record of the IBM credentials on your instance. Next, you create the service account that is associated with the credentials.

12. Navigate to **Cloud Management > Service Accounts**.
13. Click **New**, enter a unique and meaningful **Name**, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account ID</td>
<td>Enter the IBM software user number that the instance should use to access the IBM Cloud Console.</td>
</tr>
<tr>
<td>Discovery credentials</td>
<td>Select the credentials record that you created earlier.</td>
</tr>
<tr>
<td>Datacenter URL</td>
<td>Leave the <strong>Datacenter URL</strong> field blank.</td>
</tr>
<tr>
<td>Datacenter type</td>
<td>Select <strong>IBM Datacenter (cmdb_ci_ibm_datacenter)</strong>.</td>
</tr>
<tr>
<td>Datacenter discovery status</td>
<td>Auto-generated value: Status and timestamp of the last execution of Discovery on the datacenter.</td>
</tr>
<tr>
<td>Should pull events</td>
<td></td>
</tr>
</tbody>
</table>

14. Click **Update** or **Submit**.

The system creates the service account and displays the list of all discovered datacenters.

15. Repeat the process to add as many service accounts as needed.

**Day 1 setup guide for VMware on Cloud Management**

To set up Cloud Management for the very first time, you perform the procedures in this "Day 1" setup guide. Be sure to perform the procedures in order. After you have performed Day 1 setup, you can perform optional Day 2 setup and configuration procedures as needed and in any order. Detailed instructions for each procedure follow this overview.

**Request the Cloud Management application**

The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt). See [Request the Cloud Management application](#).

**Roles required to set up Cloud Management**

- Operations in the VMware Console require the VMware administrator role.
- Operations in Cloud Management require the sn_cmp.cloud_admin role.

**About terms that Cloud Management uses**

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings.

**Quick overview of the setup process**

Your setup process includes these tasks:

- If needed: Request the Cloud Management application.
• Assign appropriate roles to cloud users.
• Set up the MID Servers that will handle secure communications with the provider API endpoints.
• On the provider portal, collect your account settings and the credentials that the Discovery process will use (through a MID Server) to programatically access your provider accounts. Securely associate the account settings and credentials with a service account in Cloud Management.
• Set up a cloud account to represent your entire managed cloud infrastructure and set up a service account that works with one of your provider accounts. You specify which datacenters in the service account should be included in the cloud account. (Later, on ‘Day 2’, you can set up additional cloud accounts and service accounts from the same or other providers.)
• Cloud Management uses CMDB data to help users request and manage cloud resources and to help you manage your cloud infrastructure. To populate the CMDB with resource data for all datacenters, you manually run the Discovery process on each datacenter in the service account. Then, to ensure that the data continues to be updated, you configure Discovery to run on a regular schedule. Your cloud account might look like this:

![MyFirst-CloudAccount Diagram]

• Providers offer services that can auto-update the CMDB whenever a create/modify/terminate life-cycle change or configuration update occurs to a resource. You can configure the service to integrate directly with Cloud Management.

---

**What you will do to integrate Cloud Management with your VMware cloud accounts**

Detailed instructions for each procedure follow this overview.

1. **Assign roles to VMware users of Cloud Management**
   You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

2. **Set up MID Servers to connect Cloud Management to a VMware service account**
   To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.

   **Note:** Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if...
3. Create the credential and service account that will access your VMware data

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you first create an account in the VMware Console. You then securely store the credentials in a service account in your instance. A service account is a secure record on your instance that stores the credential and access information for your provider account. Discovery uses the information to access your provider account to get data on each resource in each specified datacenter.

4. Discover all datacenters in a service account on-demand

Discovery uses the information in the service account to identify all logical datacenters associated with the provider account.

5. Set up cloud accounts for VMware

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts—even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

6. Set capacity limits on user requests for resources

Capacity limits place restrictions on the attributes of cloud resources such as the number of virtual machines, virtual CPUs, or aggregate storage. You can set limits on resources separately for each logical datacenter in a cloud account.

7. Configure the VMware Events service to auto-update the CMDB

The VMware Events service can auto-update CI data in the CMDB whenever Cloud Management makes a life-cycle state or configuration change to a VMware resource. As a result, the CI data in the CMDB is updated without having to wait for Discovery to run.

VMware prerequisites

**vSphere**
- Enterprise or Enterprise Plus Edition 5.5 or newer.
- Enable the vSphere environment for Distributed Resource Scheduling (DRS).
- Place datastores under Storage DRS.

When provisioning Windows virtual machines, read and modify the named customization specifications.

**ESXi hosts**

If the IP address of a virtual machine is set manually using a script or post-init scripts run on the VM, then you must allow HTTPS traffic between the MID Server and the ESXi host for the VM. Enable NTP time sync. Enable the NTP client for the hypervisor on the vSphere web client. See the [vSphere documentation](#) for more information.

**vCenter server requirements**

vCenter 5.5 or 6.0 Standard or Foundation edition.

**vCenter access requirements**

Minimum permissions for the user that connects to vSphere through the instance:
- Datastore: All
- Datastore Cluster: All
- Folder: Create, Delete
- Global: Cancel task, License
- Host > Local operations: Create/Delete/Reconfigure virtual machines
- Profile-driven storage: All
- Network: Assign network
- Resource: Apply recommendation; Create/Modify/Assign/Remove virtual machines.
- Virtual machines: All

vCenter credentials

- Username
- Password
- URL or IP address of the vCenter server that the instance access through the MID Server.
- vCenter instance UUID. See Create the credential and service account that will access your VMware data for instructions on how to obtain the UUID.

Next steps

When you have finished all Day-1 and Day-2 procedures in this setup guide, see the Cloud Management administration guide for information on using the Cloud Management application in your organization.

Assign roles to VMware users of Cloud Management

You assign Cloud Management roles to user groups and to individual users based on user activities and responsibilities.

Role required: user_admin or admin

Assign the following roles to groups and users as appropriate:

<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Description and tasks</th>
<th>Access rights in Cloud Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root administrator</td>
<td>Highest level of application access for Cloud Management.</td>
<td>All</td>
</tr>
<tr>
<td>[sn_cmp.cmp_root_admin]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cloud administrator</td>
<td>Configures the Cloud Management application and sets up the cloud infrastructure.</td>
<td>Cloud infrastructure:</td>
</tr>
<tr>
<td>[sn_cmp.cloud_admin]</td>
<td></td>
<td>- Service accounts and cloud account</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Networks and IPAMp</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Settings for provider services that auto-update the CMDB:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AWS Config</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Azure Alert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Google Stackdriver Logging</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- IBM Cloud Update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- VMware Events</td>
</tr>
<tr>
<td>Descriptive name and role name</td>
<td>Description and tasks</td>
<td>Access rights in Cloud Management</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------</td>
<td>------------------------------------</td>
</tr>
</tbody>
</table>
| Governor                     | Monitors overall cloud usage and enforces compliance of the organization's rules, quotas, and policies. Also manages tags and permissions to various objects. | Governance:  
  - Policies  
  - Pools  
  - Quotas  
  - Permissions |
| Service Designer             | Creates blueprints, ARM and CloudFormation templates, and catalog items. | Cloud Service Design:  
  - Cloud templates  
  - Blueprints  
  - Blueprint catalog items  
  - Resource blocks |
| Cloud user                   | Requests and manages stacks and resources. | Cloud User Portal  
  When you assign the role to a group, all members of the group share quota limitations and ownership of certain resources. |
| Cloud operator               | Monitors and troubleshoots the Cloud Management application. | Dashboards and reports:  
  - Cloud Operations Dashboard  
  - Cloud Root Cause Analysis  
  - Cloud Orchestration Trail  
  - Cloud API Trail |
| Cloud Event Integration      | Authorizes access to the instance for external services that auto-update the CMDB when cloud events occur. This role gives the access to the cloud event REST endpoint. | Cloud infrastructure:  
  - Service accounts and cloud account  
  - Networks and IPAM  
  Settings for provider services that auto-update the CMDB:  
    - AWS Config  
    - Azure Alert  
    - Google Stackdriver Logging  
    - VMware Events |
| Cloud Group administrator    | Grants admin access to any group that you belong to. | |

Set up MID Servers to connect Cloud Management to a VMware service account

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.
Role required: admin or sn_cmp.cloud_admin

- Data is encrypted to the MID Server and between the MID Server and the API endpoint. To ensure high performance and security, you should configure one or more MID Servers for each datacenter under management. Configure the MID Servers even if you have already configured other MID Servers while setting up Cloud Management for another cloud provider.
- You can configure the MID Server for Cloud Management to use a proxy server. See Using a proxy server for your MID Server.

1. You configure the connection to a MID Server in the same way as for any other applications that use MID Servers, like Discovery and Service Mapping. See MID Server for instructions on setting up and configuring MID Servers.

   **Note:** Do not configure the MID Server Application related list.

2. Using a proxy server for your MID Server
   You can configure the MID Server for Cloud Management to use a proxy server. Using a proxy server supports all cloud-based activities such as running Discovery, billing downloads, provisioning virtual machines, and running life-cycle operations on virtual machines.

Proxy server limitations

- Windows or Linux platforms only are supported.
- The Google Cloud Console integration does not support downloading billing data.
- VMware is not supported.
- Remote PowerShell scripts cannot be executed.
- Custom APIs may not work.

Supported proxy server configurations for Cloud Management

<table>
<thead>
<tr>
<th>Authentication/No Authentication</th>
<th>Authentication type</th>
<th>Proxy Server type</th>
</tr>
</thead>
<tbody>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>No authentication</td>
<td>No authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>HTTP/HTTPS</td>
</tr>
<tr>
<td>Authentication</td>
<td>Basic authentication</td>
<td>SOCKS5</td>
</tr>
<tr>
<td>Authentication</td>
<td>NTLM</td>
<td>HTTP/HTTPS</td>
</tr>
</tbody>
</table>

Qualified proxy server configurations for Cloud Management

<table>
<thead>
<tr>
<th>Qualified configuration settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration</td>
</tr>
<tr>
<td>---------------</td>
</tr>
<tr>
<td>Configuration 1</td>
</tr>
<tr>
<td>Configuration 2</td>
</tr>
</tbody>
</table>
### ServiceNow New York IT Operations Management

#### Configuration
<table>
<thead>
<tr>
<th>Configuration</th>
<th>Operating system</th>
<th>Proxy server</th>
<th>Authentication mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration 3</td>
<td>Linux</td>
<td>Squid (HTTPS)</td>
<td>Local</td>
</tr>
<tr>
<td>Configuration 4</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>Active Directory</td>
</tr>
</tbody>
</table>

#### Qualified configurations

<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS, Azure, and IBM Cloud</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Google Cloud Platform</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing download</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>VMware</td>
<td>Not Applicable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Create the credential and service account that will access your VMware data

To securely access data on your provider account, the Discovery process must present appropriate credentials. To make the credentials available to Discovery, you first create an account in the VMware Console. You then securely store the credentials in a service account in your instance.

Roles required:
- VMware Console administrator
- Cloud Management: admin or sn_cmp.cloud_admin
Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings.

1. In the Cloud Admin Portal, navigate to Manage > Credentials.
2. On the Credentials tab, click New, select VMware Credentials, fill in the form, and then click Submit.

### VMware Credentials form fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique and meaningful name for the credentials (for example, Cloud Management VMware Credentials).</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to enable the credential for use.</td>
</tr>
<tr>
<td>User name</td>
<td>User name for access to your vSphere account.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for access to your vSphere account.</td>
</tr>
</tbody>
</table>

3. Navigate to Cloud Management > Service Accounts, click New, enter a unique and meaningful Name (for example, Cloud Management Service Account), and then fill in the form.

### New service account for VMware

<table>
<thead>
<tr>
<th>Account ID</th>
<th>There are two methods for determining the value:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• If you have access to the managed object browser for the vCenter (MOB): The Account ID is the instance UID. Open the MOB and then click Content &gt; About. Paste the value of the instanceUuid property in the Account ID field.</td>
</tr>
<tr>
<td></td>
<td>• If you do not have access to the MOB, enter the temporary text &quot;vmware-temp-1&quot; in the Account ID field. In a step that follows this table, you create the actual value and then return to this form to replace the temporary text with the actual value.</td>
</tr>
<tr>
<td>Discovery credentials</td>
<td>Name of the credentials that you created in the preceding step. Cloud Management VMware Credentials in the example.</td>
</tr>
<tr>
<td>Datacenter URL</td>
<td>Full URL of the vCenter, for example: https://&lt;IP address&gt;/sdk.</td>
</tr>
<tr>
<td>Datacenter Type</td>
<td>Select VMware vCenter datacenter (cmdb_ci_vcenter_datacenter).</td>
</tr>
<tr>
<td>Datacenter discovery status</td>
<td>Auto-generated value: Status and timestamp of the last execution of Discovery on the datacenter.</td>
</tr>
<tr>
<td>Should pull events</td>
<td>Enables Cloud Management to collect events from resource types in VMware vCenter datacenter in the sn_cmp_cloud_event_list.</td>
</tr>
</tbody>
</table>

4. Click the Discover Datacenters related link. Using the credential, Discovery accesses the vCenter at the specified URL to discover VMware datacenters.

5. If you used the instance UUID value in the Account ID field, click Submit and skip the remaining steps in this procedure.

6. On the Cloud Admin Portal, navigate to Operate > Trails.

7. On the Cloud API Trail tab, find the entry with the Method name value of ListDatacenters and Provider name value of vmware-compute (typically the most-recent entry). Open the entry.

8. On the CAPI Trail Logs related list, open the route_result entry.

9. In the Log value text box, copy the locationId value. This value is the Account ID that you will paste into the form in the next step.

10. On the instance, navigate to Cloud Management > Service Accounts and then click the service account that you created earlier (Cloud Management Service Account in the example).

11. Paste the locationId value (that you copied from the CAPI Trail Log form) in the Account ID field on the form and then click Update.
12. Click **Submit**.
   The system creates the service account and displays the list of all discovered datacenters.

**Discover all datacenters in a service account on-demand**

After you set up a service account, run Discovery to populate the datacenters in the account.

- Role required: sn_cmp.cloud_admin
- A cloud account must exist with service accounts and associated datacenters. You must run on-demand Discovery to discover datacenters before you can schedule Discovery.
- Download the Discovery and Service Mapping pattern from the ServiceNow Store.

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In the Google Cloud Platform, the region-specific containers for virtual resources are called **regions**. In Cloud Management, regions are called datacenters or logical datacenters (LDCs). The term "logical" is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.
You can run Discovery on demand without waiting for the next scheduled run. See Schedule Discovery of all resources in a datacenter.

1. Navigate to **Cloud Management > Service Accounts**.
2. Click the name of a service account and then click the **Discover Datacenters** related link.
3. Wait for the **Datacenter discovery status** field to show a status of **Complete**.
   - If Discovery is successful, you see the following message in the **Datacenter discovery status** field: `YYYY-MM-DD HH:MM:SS: Completed ...`. **Reload this form to see an updated datacenter list.**
4. **Reload the form.**

The discovered datacenters appear in the Logical Datacenters related list.

**Note:** Check the **Cloud API Trail** and the **Orchestration Trail** if Discovery encounters issues.

### Set up cloud accounts for VMware

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

**Role required:** `sn_cmp.cloud_admin`

See **Cloud accounts** for detailed information.

1. In the Cloud Admin Portal, navigate to **Manage > Cloud Accounts**.
2. Click **New**, enter a unique and meaningful **Name** and **Description**, and then select the cloud **Provider** for the account (VMware vCenter).
3. Click **Next**.
Based on the provider you select, the **Service Account** field is populated and a list of discovered datacenters appears. If no datacenters appear, click **Discover Datacenters** to find them. See *Discover all datacenters in a service account on-demand* for details.

4. Select one or more datacenters to associate with the cloud account, click **Save**, and then close the popup.
   The Cloud Account dashboard appears. The datacenters that you selected appear on the **Datacenters** tab.

5. Click a datacenter.
   The following lists appear:
   • **Resources**: Cloud resources for the current datacenter. Run Discovery to populate the CMDB for the datacenter and populate the tab. See *Discover all datacenters in a service account on-demand*.
   • **Capacity Limits**: Limits on virtual CPUs, virtual networks, storage volume size, and other services. See *Set capacity limits on user requests for resources*.
   • **Discovery Log**: The process creates a log record for each action in an datacenter.

Repeat the process to create additional cloud accounts as needed. Run Discovery and set capacity limits for each datacenter.

*Add a datacenter to a cloud account*
At any time, you can add a logical datacenter to the cloud infrastructure that is represented by a cloud account.
Role required: sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to Manage > Cloud Accounts.
2. Open a cloud account, set the state to Draft, and then click Configure in the form header.
3. Click the Datacenters tab and then select the Service Account that holds the credentials that enable access to the provider account that includes the datacenter to add.

When you select the service account, the list of discovered datacenters for the service account appears.

<i>Note: </i>If the expected datacenters do not appear, click Discover Datacenters to update the list. Discovery runs and displays all datacenters associated with the service account.

4. Select the datacenters to add and then click Save.

In the example, you select <i>ap-northeast-2</i>.
The datacenters are added to the cloud account and appear on the **Datacenters** tab. When Discovery runs, the resources in the datacenter appear on the **Resources** tab.

![Datacenters Image]

**Configure the VMware Events service to auto-update the CMDB**

The VMware Events service can auto-update CI data in the CMDB whenever Cloud Management makes a life-cycle state or configuration change to a VMware resource. As a result, the CI data in the CMDB is updated without having to wait for Discovery to run.

- **Role required:** sn_cmp.cloud_event_integration
- **You must have** a cloud account with VMware service accounts and associated logical datacenters.
- **The following Cloud Management plugins must be activated:**
  - Cloud Management (com.snc.cloud.mgmt)
  - Cloud Management Core (com.snc.cloud.core)
  - Cloud API (com.snc.cloud.api)
  - Cloud Config Management (com.snc.config.mgmt)
- **A MID server needs to be available with VMware capabilities.**
- **The Vcenter datacenters are successfully discovery and associated data is populated.**

When VMware Events sends an update, the instance processes the event and creates or updates the CI entry in the CMDB and the CI information in the Cloud User Portal. Each event is saved as a record in the Events table on your instance. Configure all event/alert rules from the vCenter.

**Note:** To connect to a different MID Server or vCenter: Update the settings and then click the **Start** related link. The **Status** value changes to **Updating** and then to **Started**.

1. On the **Cloud Admin Portal**, navigate to **Manage > Alert Configurations**.
2. On the **VMware** tab, click **New**, enter a unique and meaningful **Name** and **Short description**, and then fill in the form:

<table>
<thead>
<tr>
<th>Execute on</th>
<th>Select <strong>Specific MID Server</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>MID Server</td>
<td>Select a MID Server.</td>
</tr>
</tbody>
</table>
3. Right-click in the header and select **Save**.

4. Test the connection between the vCenter and the MID Server: Click **Test parameters**. A pop-up displays **Parameters verified** for success. The **Status** field does not change. To update a parameter (for example, to change the MID Server or vCenter), change the setting and then click **Update parameters**.

5. Click the **Start** related link to start collecting events for vCenter.  
   - To confirm that the instance is receiving events, verify that the **Status** field displays **Started**.  
   - Click **Stop** to stop collecting events for a particular vCenter.  
   - Click **Restart** to restart event collection.

vCenter events are being populated in **sn_cmp_cloud_event** table.  
- If the vCenter has events in the Event Manager and the same events are configured, Cloud Management gathers the events and take action on the CIs accordingly.  
- Identification and Reconciliation Engine (IRE) changes the CI state based on event payload received and the IRE cannot be customized.

### Additional Cloud Management setup on day 2

After you have performed Day 1 setup, you can perform optional setup and configuration procedures as needed and in any order. Detailed instructions for each procedure follow this overview.

#### Optional setup and configuration operations

Detailed instructions for each procedure follow this overview.

#### Set up an additional cloud account

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

#### Add a datacenter to a cloud account

At any time, you can add a logical datacenter to the cloud infrastructure that is represented by a cloud account.

#### Schedule discovery of all resources in a service account

You can add a custom schedule for Discovery to discover all resources each datacenter in a service account and then update the CMDB with the new information.

#### Schedule Discovery of all resources in a datacenter

<table>
<thead>
<tr>
<th>vCenter</th>
<th>Select the vCenter to monitor.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extension (read only)</td>
<td>The MID Server extension of this context. Updated when data collection starts.</td>
</tr>
<tr>
<td>Status (read only)</td>
<td>The status of the data collection process, for example, Starting, Started, and so on.</td>
</tr>
<tr>
<td>Executing on (read only)</td>
<td>The MID Server that is executing the event collector</td>
</tr>
</tbody>
</table>
To update a particular datacenter more frequently than is scheduled for the cloud account, you can add a custom schedule for discovering all resources in the datacenter. To view information about a discovered resource, select a datacenter on the Logical Datacenters related list and then click the resource.

**Discover all resources in a datacenter on-demand**

At any time, you can run Discovery on a datacenter to update the CMDB.

**Discover all datacenters in a service account on-demand**

After you set up a service account, run Discovery to populate the datacenters in the account. Discovery uses the information in the service account to identify all logical datacenters associated with the provider account.

**Set capacity limits on user requests for resources**

Capacity limits place restrictions on the attributes of cloud resources such as the number of virtual machines, virtual CPUs, or aggregate storage. You can set limits on resources separately for each logical datacenter in a cloud account.

**Next steps**

When you have finished all Day-1 and Day-2 procedures in this setup guide, see the Cloud Management administration guide for information on using the Cloud Management application in your organization.

**Set up an additional cloud account**

During initial installation, you set up one cloud account. To organize and compartmentalize your infrastructure, you can set up additional cloud accounts to include different providers or service accounts or datacenters.

Role required: sn_cmp.cloud_admin

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts— even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account. See Cloud accounts for detailed information.

1. In the Cloud Admin Portal, navigate to Manage > Cloud Accounts.
2. Click New, enter a unique and meaningful Name and Description, select the cloud Provider for the account (MyProvider in the example), and then click Next.
Based on the provider you select, the **Service Account** field is populated.

3. Select the appropriate **Service Account**.
   A list of discovered datacenters appears. If no datacenters appear, click **Discover Datacenters**. See *Discover all datacenters in a service account on-demand* for details.

4. Select each of the datacenters that you want to associate with the cloud account, click **Save**, and then close the popup.

   The datacenters appear on the **Datacenters** tab for the cloud account.
5. When the datacenters appear, select one or more datacenters to include in the cloud account and then click Save. The Cloud Account dashboard appears. The datacenters that you selected appear on the Datacenters tab.

6. Click a datacenter. The following lists appear:
   - Resources: Cloud resources for the current datacenter. Run Discovery to populate the CMDB for the datacenter and populate the tab. See Discover all datacenters in a service account on-demand.
   - Capacity Limits: Limits on virtual CPUs, virtual networks, storage volume size, and other services. See Set capacity limits on user requests for resources.
   - Discovery Log: The process creates a log record for each action associated with a discovery status.

7. Repeat the process to add as many service accounts as needed.

Repeat the process to create additional cloud accounts as needed. Run Discovery and set capacity limits for each datacenter.

Add a datacenter to a cloud account

At any time, you can add a logical datacenter to the cloud infrastructure that is represented by a cloud account.

Role required: sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to Manage > Cloud Accounts.
2. Open a cloud account, set the state to Draft, and then click Configure in the form header.
3. Click the Datacenters tab and then select the Service Account that holds the credentials that enable access to the provider account that includes the datacenter to add.
When you select the service account, the list of discovered datacenters for the service account appears.

**Note:** If the expected datacenters do not appear, click Discover Datacenters to update the list. Discovery runs and displays all datacenters associated with the service account.

4. Select the datacenters to add and then click **Save**.
   
   In the example, you select **ap-northeast-2**.

   The datacenters are added to the cloud account and appear on the **Datacenters** tab. When Discovery runs, the resources in the datacenter appear on the **Resources** tab.
Schedule discovery of all resources in a service account

You can add a custom schedule for Discovery to discover all resources each datacenter in a service account and then update the CMDB with the new information.

- Role required: sn_cmp.cloud_admin
- A cloud account must exist with service accounts and associated datacenters. You must run on-demand Discovery to discover datacenters before you can schedule Discovery.
- Download the Discovery and Service Mapping pattern from the ServiceNow Store.

- Scheduled Cloud Discovery discovers only datacenters that are part of the service account when the schedule is created. If datacenters have been added since the time you created the schedule, Discovery does not discover those datacenters. Delete the old schedule and create a new schedule every time the datacenters in a service account change.
- Discovery populates the CMDB with the discovered resources and updates the resource information on the Resources tab. You can view Discovery results on the Discovery Log tab.

1. In the Cloud Admin Portal, navigate to Manage > Discovery Schedules.
2. Click Add Cloud Schedule.
3. On the Add Account page, select the cloud Provider and enter a unique, descriptive Name for the schedule.
4. In the Service Account section, select Select Account, specify the service account, and then click Next.
   
   **Note:** To create a service account, select Add Account and then fill in the form. For details, see the process of setting up a service account in the Day 1 setup guide.

5. On the Select Datacenters page, you specify which of the datacenters in the service account to discover. Move the datacenters to discover into the Selected list or select Discover all datacenters listed below to move all datacenters to the Selected list. Click Next.

6. On the Create Schedule page, you specify the time and frequency of discovery.
Note: Run time setting uses the system time zone.

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>On Demand</td>
<td>Does not run on a schedule. Click Discover now to run Discovery. This is the default option.</td>
</tr>
<tr>
<td>Daily</td>
<td>Runs every day. Use the Start time field to specify the time of day.</td>
</tr>
<tr>
<td>Weekly</td>
<td>Runs on one designated day of each week. Use the Start time field to specify the time of day.</td>
</tr>
<tr>
<td>Monthly</td>
<td>Runs on one designated day of each month. Use the Day field to select the day, specify the time of day, if the designated day does not occur in the month, for example, if you designate day 30, the schedule does not run in February.</td>
</tr>
<tr>
<td>Periodically</td>
<td>Runs every designated period of time. Use the Repeat Interval field to define how often the schedule runs. The first Discovery runs at the point in time defined in the Starting field. The subsequent discoveries run after each Repeat Interval period passes.</td>
</tr>
<tr>
<td>Once</td>
<td>Run one time as designated by the date and time defined in the Starting field.</td>
</tr>
<tr>
<td>Weekdays</td>
<td>Runs every Monday, Tuesday, Wednesday, Thursday, and Friday. Use the Start time field to select the time of day.</td>
</tr>
<tr>
<td>Weekends</td>
<td>Runs every Saturday and Sunday. Use the Start time field to select the time of day.</td>
</tr>
<tr>
<td>Month Last Day</td>
<td>Run the last day of every month. Use the Start time field to select the time of day.</td>
</tr>
<tr>
<td>Calendar Quarter End</td>
<td>Runs on March 31, June 30, September 30, and December 31. Use the Start time field to select the time of day. To change the dates, modify the DiscoveryScheduleRunType script include.</td>
</tr>
<tr>
<td>After Discovery</td>
<td>Allows you to sequentially stagger the schedule. Use this option to run this schedule after the Discovery designated in the Run after field finishes. Select the Even if canceled check box to designate that this discovery should run even if the Discovery is canceled before it finishes.</td>
</tr>
<tr>
<td></td>
<td>• This option is not valid when the Discovery is started via DiscoverNow, or when you cannot designate an inactive Discovery schedule.</td>
</tr>
<tr>
<td></td>
<td>• You cannot create a loop by designating the run after Discovery to be the same Discovery.</td>
</tr>
<tr>
<td></td>
<td>• This Discovery does not run if the Run after Discovery does not finish, with the exception that the Even if canceled check box is selected and the Discovery is canceled.</td>
</tr>
<tr>
<td>Cancel Discovery if longer than Time limit for running the discovery. When the specified time elapses, the remaining tasks for the discovery are canceled even if the discovery is not finished. If the check box is not selected, the discovery runs until finished. This setting constrains system load to a desirable time window.</td>
<td></td>
</tr>
</tbody>
</table>

7. Click Finish and Run Now.

Discovery runs immediately. Discovery runs at the specified times and then adds discovered datacenters to the Logical Datacenters related list.

If you configured a Discovery schedule, it appears in the Discovery Config - Logical Datacenters related list. To run Discovery on a single datacenter, add the datacenter to the Discovery schedule in the related list, then run Discovery again.

Open a datacenter record to see the relationships between the datacenter and the resources in the datacenter.
To view discovery status, discovery logs, discovered devices, and the ECC queue, click **Advanced View**.

If Discovery fails, check the Cloud Discovery results on the [home page](#).
Schedule Discovery of all resources in a datacenter

To update a particular datacenter more frequently than is scheduled for the cloud account, you can add a custom schedule for discovering all resources in the datacenter. To view information about a discovered resource, select a datacenter on the Logical Datacenters related list and then click the resource.

- Role required: sn_cmp.cloud_admin
- A cloud account must exist with service accounts and associated datacenters. You must run on-demand Discovery to discover datacenters before you can schedule Discovery.
- Download the Discovery and Service Mapping pattern from the ServiceNow Store.

- Scheduled Discovery discovers only datacenters that are part of the service account when the schedule is created. If datacenters have been added since the time you created the schedule, Discovery does not discover those datacenters. Delete the old schedule and create a new schedule every time the datacenters in a service account change.
- You can run Discovery on-demand without waiting for the next scheduled Discovery. See Discover all resources in a datacenter on-demand.

To view the list of defined schedules:

1. In the Cloud Admin Portal, navigate to Manage > Cloud Accounts.
2. Open the cloud account that includes the service account with the datacenter and then click the Schedules tab.
3. Select a schedule to view details.

1. Navigate to Cloud Management > Service Accounts.
2. Click a service account and, on the Discovery Config - Logical Datacenters related list, click New.
3. On the Discovery Jobs tab, click Schedule Discovery and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery schedule</td>
<td>Select the schedule for Discovery.</td>
</tr>
<tr>
<td>Logical datacenter</td>
<td>Select the datacenter to discover on the schedule.</td>
</tr>
<tr>
<td>Service account</td>
<td>The service account that you selected appears in the field. You can select a different service account.</td>
</tr>
</tbody>
</table>

4. Select the Active check box to activate the schedule and then click Submit.

Discover all datacenters in a service account on-demand

After you set up a service account, run Discovery to populate the datacenters in the account.

- Role required: sn_cmp.cloud_admin
- A cloud account must exist with service accounts and associated datacenters. You must run on-demand Discovery to discover datacenters before you can schedule Discovery.
- Download the Discovery and Service Mapping pattern from the ServiceNow Store.

Cloud providers often use proprietary names for account and credential settings. Because the Cloud Management application supports several cloud providers, Cloud Management uses general-purpose names for the settings. In the Google Cloud Platform, the region-specific...
containers for virtual resources are called regions. In Cloud Management, regions are called datacenters or logical datacenters (LDCs). The term “logical” is used to reinforce the idea that Cloud Management is provider-agnostic. All infrastructure or applications that are deployed using Cloud Management are associated with a datacenter.

- You can run Discovery on demand without waiting for the next scheduled run. See Schedule Discovery of all resources in a datacenter.

1. Navigate to Cloud Management > Service Accounts.
2. Click the name of a service account and then click the Discover Datacenters related link.
3. Wait for the Datacenter discovery status field to show a status of Complete. If Discovery is successful, you see the following message in the Datacenter discovery status field: YYYY-MM-DD HH:MM:SS: Completed …. Reload this form to see an updated datacenter list.
4. Reload the form.

The discovered datacenters appear in the Logical Datacenters related list.

**Note:** Check the Cloud API Trail and the Orchestration Trail if Discovery encounters issues.

**Discover all resources in a datacenter on-demand**

At any time, you can run Discovery on a datacenter to update the CMDB.

- Role required: sn_cmp.cloud_admin
- A cloud account must exist with service accounts and associated datacenters. You must run on-demand Discovery to discover datacenters before you can schedule Discovery.
- Download the Discovery and Service Mapping pattern from the ServiceNow Store.

1. In the Cloud Admin Portal, navigate to Manage > Cloud Accounts.
2. Click a cloud account and, on the Datacenters tab, select a datacenter.
   The Resources tab lists the resources in the CMDB.

3. Change the state to **Draft** and then click Discover Now. Wait for Discovery to finish.
4. Change the state of the cloud account to Published.

Discovery populates the CMDB with the discovered resources and updates the information on the Resources tab. You can view Discovery results in the Discovery Log tab.

**Set capacity limits on user requests for resources**

Capacity limits place restrictions on the attributes of cloud resources such as the number of virtual machines, virtual CPUs, or aggregate storage. You can set limits on resources separately for each logical datacenter in a cloud account.

- Role required: sn_cmp.cloud_admin
- A cloud account must exist with service accounts and associated datacenters. You must run on-demand Discovery to discover datacenters before you can schedule Discovery.
- Download the Discovery and Service Mapping pattern from the ServiceNow Store.

**Note:** You can also use quotas to set per-user and total capacity limitations on resources. See [Quota governance for Cloud Management](#).

1. Navigate to **Cloud Management > Cloud Admin Portal > Manage > Cloud Accounts**.
2. Click a cloud account, and then, on the **Datacenters** tab, select a datacenter.
3. Set the state to **Draft**.
4. On the **Capacity Limits** tab, set the **Capacity Limit** values as needed.
<table>
<thead>
<tr>
<th>Name</th>
<th>Discovered</th>
<th>Provisioned</th>
<th>Capacity Limit</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Count</td>
<td>11</td>
<td>1</td>
<td>100</td>
<td>Single</td>
</tr>
<tr>
<td>Storage Size</td>
<td>57</td>
<td>0</td>
<td>3000</td>
<td>GB</td>
</tr>
<tr>
<td>VM Count</td>
<td>4</td>
<td>2</td>
<td>1000</td>
<td>Single</td>
</tr>
<tr>
<td>vCPUs Count</td>
<td>4</td>
<td>2</td>
<td>2000</td>
<td>Single</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovered</td>
<td>Total value discovered by the Discovery process for all resources in the selected service account.</td>
</tr>
<tr>
<td>Provisioned</td>
<td>Total value provisioned for the selected service account.</td>
</tr>
<tr>
<td>Capacity Limit</td>
<td>Maximum allowed value calculated as the sum for all resources in the selected service account.</td>
</tr>
<tr>
<td>Unit</td>
<td>Unit of measure for the specified value. For example, storage is measured in units of GB.</td>
</tr>
</tbody>
</table>

5. Click **Update**.
6. Set the state to **Published**.

Click **Discover Now** to run Discovery on the logical datacenters.
Cloud Management administration guide

After you set up the Cloud Management application on your instance, there are many items that you must configure before your users can request and manage any cloud resources.

Requirements

Role required: cloud_admin

About Cloud Management

Before you begin

Set up Cloud Management

See the appropriate setup guide for your cloud provider:

- Cloud Management setup guide for AWS
- Cloud Management setup guide for Azure
- Cloud Management setup guide for VMware

What to do after setup

Create resource profiles

Resource profiles contain the attributes that a resource can have. Users select values for these attributes when they order a cloud resource. Configure these types of resource profiles: Compute (such as the size of a VM), Operating System (such as the type and version), Storage (such backup) and Security Groups (such as whether or not to use HTTPS). See Resource profiles for more information.

Configure governance

Governance tools that you can use to restrict the provisioning of cloud resources include quotas and policies. Examples of things you can configure include the approval workflow that is used after a user requests a cloud resource, whether or not an email is sent after a cloud resource is stopped, and which datacenter should be automatically selected when a user selects a resource from a specific cloud provider. See Cloud Management governance for more information.

Populate the Cloud Service Catalog

After you apply governance quotas and policies, create Service Catalog items. You can create a catalog item in the following ways:

- Import a template from AWS or Azure
- Use or modify a base-system catalog item
• Generate a catalog item from a blueprint. A blueprint is a configurable template for a catalog item.

See [Blueprints](#) for more information.

**Domain separation in Cloud Management**

This is an overview of domain separation in Cloud Management. Domain separation enables you to separate data, processes, and administrative tasks into logical groupings called domains. You can then control several aspects of this separation, including which users can see and access data.

**Overview**

Domain separation is not supported in this application. For more information, see [Application support for domain separation](#).

**Cloud Management Recipes**

Multi-cloud recipes provide ready content for typical cloud deployment and common operations scenarios across cloud platforms.

**Recipes for Enterprise Workloads and Day-2 Operations**

Cloud Management recipes are ready-to-use catalog items built from sample CFT and ARM templates to support real-world cloud deployment use-cases. These recipes are ready for use in Proof of Concept as well as non-production environments. For production environments, ensure that the recipes are well tested and internally evaluated. You can avail the recipes, as update sets, to be downloaded from the SHARE in the developer portal under the ‘Cloud Management Recipes’ category.

**List of Update Sets**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog Items for Amazon AWS</td>
<td></td>
</tr>
<tr>
<td>CFT Single Linux VM</td>
<td>Catalog Item for a single Linux Virtual Machine using CFT.</td>
</tr>
<tr>
<td>CFT Single Windows VM</td>
<td>Catalog Item for a single Windows Virtual Machine using CFT.</td>
</tr>
<tr>
<td>CFT Linux VM with Multiple NICs</td>
<td>Catalog Item for a Linux Virtual Machine with multiple network interfaces using CFT.</td>
</tr>
<tr>
<td>CFT Linux VM with LB and Storage</td>
<td>Catalog Item for a Linux Virtual Machine with load balancing and storage using CFT.</td>
</tr>
<tr>
<td>CFT Linux VM with Multi VM Multi Disk</td>
<td>Cloud Formation Template for a Linux VM with multiple VMs and multiple virtual disk interfaces.</td>
</tr>
<tr>
<td>CFT Linux Install Apache</td>
<td>Catalog Item to install an Apache server on a Linux VM using CFT.</td>
</tr>
<tr>
<td>CFT Simple VM with Approval Policy</td>
<td>Catalog Item for a simple VM with an approval policy, using CFT.</td>
</tr>
</tbody>
</table>
### Catalog Items for Azure

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFT Windows VM Multi Disk and LB</td>
<td>Catalog Item for a Windows VM with multiple virtual disk interfaces and load balancing, using CFT.</td>
</tr>
<tr>
<td>ARM Windows VM with Multi NiCs</td>
<td>Catalog Item for a Windows VM with multiple network interfaces using ARM template.</td>
</tr>
<tr>
<td>ARM Windows Single VM Multi Disk</td>
<td>Catalog Item for a single VM with multiple virtual disk interfaces, using ARM template.</td>
</tr>
<tr>
<td>ARM Windows Stack Multi Disk</td>
<td>Catalog Item for a Windows stack with multiple virtual disk interfaces, using ARM template.</td>
</tr>
<tr>
<td>ARM Windows Stack Multi NIC</td>
<td>Catalog Item for a Windows Stack VM with multiple network interfaces, using ARM template.</td>
</tr>
<tr>
<td>ARM Linux VM Multi NIC Multi Disk</td>
<td>Catalog Item for a Linux VM with multiple virtual disk and network interfaces, using ARM template.</td>
</tr>
<tr>
<td>ARM Linux VM Install Apache</td>
<td>Catalog Item to install an Apache server on a Linux VM, using ARM template.</td>
</tr>
<tr>
<td>ARM Linux VM Install MySQL</td>
<td>Catalog Item to install a MySQL server on a Linux VM, using ARM template.</td>
</tr>
<tr>
<td>ARM WebApp MySQL</td>
<td>Catalog Item to build a Web app with Azure database for MySQL, using ARM template.</td>
</tr>
<tr>
<td>ARM VNET Multi Subnet with Approval Policy</td>
<td>Catalog Item for multi subnet VNET with approval policy, using ARM template.</td>
</tr>
<tr>
<td>ARM VNIC Multi Subnet with Naming Policy</td>
<td>Catalog Item for multi subnet VNIC with a naming policy, using ARM template.</td>
</tr>
</tbody>
</table>

### Day 2 Operations for AWS VM

An update set is provided for some of the important Day 2 operations on AWS cloud provider. Broadly, Day 2 operations are the type of operations that are executed on workloads after they are provisioned. Day 2 operations can include complex life-cycle processes or simple utilities. In this update set, you have access to the following set of operations:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttachVolume</td>
<td>Attaches and exposes an Amazon (Elastic Block Store) EBS volume to the instance with the specified device name.</td>
</tr>
<tr>
<td>CreateSnapshot (AWS datacenter level discovery operation)</td>
<td>Creates a snapshot of your Amazon (Elastic Block Store) EBS volume.</td>
</tr>
<tr>
<td>DescribeSnapshots</td>
<td>Describes the specified Amazon (Elastic Block Store) EBS snapshots available to you or all EBS snapshots available to you.</td>
</tr>
<tr>
<td>DetachVolume</td>
<td>Detaches an EBS volume from an instance.</td>
</tr>
<tr>
<td>ModifyInstanceAttribute (Only InstanceType)</td>
<td>Modify the specified attribute of the specified instance.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ModifyVolume (Only Size)</td>
<td>Modify parameters of an existing Amazon (Elastic Block Store) EBS volume.</td>
</tr>
<tr>
<td>Day 2 Operations for AWS VM – Set 1</td>
<td>Catalog Item for Day 2 operations for an AWS VM, Set 1.</td>
</tr>
</tbody>
</table>

**Cloud accounts**

A cloud account is the logical representation in Cloud Management of all or part of your managed cloud infrastructure. A cloud account can include multiple service accounts — even service accounts from different providers. For each service account, you specify which datacenters to include in the cloud account.

From a cloud account, you can:

- Add LDCs. A logical datacenter (LDC) is a region-specific virtual cloud that is associated with a service account. The datacenter hosts your cloud resources.
- Run Discovery on LDCs to update the CMDB with configuration changes or life cycle changes (create/modify/terminate) for each resource in each logical datacenter (LDC) that is associated with the cloud account.
- Set capacity limits on cloud services like virtual machines, virtual CPUs, virtual networks, aggregate storage volume size, and others. Set limits to help ensure that cloud resources are provisioned at appropriate scales. Limits are especially important for capacity-constrained environments like private clouds.
- Publish a cloud account to enable business teams to deploy stacks.

**Structure of a cloud account**

An advantage of cloud accounts is that you can group specified service accounts with only the regions (datacenters) that you want to allow cloud resources in. For example, your AWS account could have more than a dozen datacenters globally. However, you might use only one or two regions. To conform with your security and compliance rules, you might not want any resources provisioned to additional regions around the world.
A cloud account can contain as many service accounts as necessary in the following scenarios:

**Possible cloud account structures**

<table>
<thead>
<tr>
<th>Structure</th>
<th>Supported</th>
<th>Not supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add the same service account as many times as necessary, provided each entry is associated with a different LDC.</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Add the same LDC from different service accounts in one cloud account.</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Add different LDCs from different service accounts to a particular cloud account, provided the regions are not the same. Example: You have us-west-1 in serviceaccount1 and us-west-2 in serviceaccount2. Both can be added to cloudaccount1. However, both us-west-1 in serviceaccount1 and us-west-1 in serviceaccount2 cannot be added to cloudaccount1.</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
Update a cloud account

You keep a cloud account in the **Draft** state until you are ready to make the datacenters and capacity limits available for use — in blueprints and the Cloud User Portal, for example. You can change the state to **Published** after you run Discovery.

Set up an additional cloud account

See [Set up an additional cloud account](#).

Add a datacenter to a cloud account

At any time, you can add a logical datacenter to the cloud infrastructure that is represented by a cloud account. See [Add a datacenter to a cloud account](#).

Set capacity limits on user requests for resources

Capacity limits place restrictions on the attributes of cloud resources such as the number of virtual machines, virtual CPUs, or aggregate storage. You can set limits on resources separately for each logical datacenter in a cloud account.

**Set capacity limits on user requests for resources**

**Service categories discovered in AWS**

The Discovery process discovers AWS services.

**Services categories discovered in AWS**

<table>
<thead>
<tr>
<th>Service Category</th>
<th>Name</th>
<th>CI Class</th>
<th>Pattern</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security</td>
<td>AWS::ACM::Certificate</td>
<td>cmdb_ci_aws_acm_certificate</td>
<td>AWS Certificate Manager</td>
<td>AWS Certificate Manager</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::CloudFormationStack</td>
<td>cmdb_ci_aws_cloudformation_stack</td>
<td>AWS CloudFormation</td>
<td>AWS CloudFormation</td>
</tr>
<tr>
<td>Tools</td>
<td>AWS::CloudTrail</td>
<td>cmdb_ci_aws_cloudtrail</td>
<td>AWS CloudTrail</td>
<td>AWS CloudTrail</td>
</tr>
<tr>
<td>Tools</td>
<td>AWS::CloudWatch</td>
<td>cmdb_ci_aws_cloudwatch</td>
<td>AWS CloudWatch</td>
<td>AWS CloudWatch</td>
</tr>
<tr>
<td>Tools</td>
<td>AWS::Config</td>
<td>cmdb_ci_aws_datacenter</td>
<td>AWS Config</td>
<td>AWS Config</td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::CustomerGateway</td>
<td>cmdb_ci_aws_customer_gateway</td>
<td>AWS Virtual Private Cloud</td>
<td>AWS Virtual Private Cloud</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::EC2::EIP</td>
<td>cmdb_ci_aws_ec2_instance_address</td>
<td>AWS Elastic Compute Cloud</td>
<td>AWS Elastic Compute Cloud</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::EC2::Image</td>
<td>cmdb_ci_aws_os_template</td>
<td>AWS Elastic Compute Cloud</td>
<td>AWS Elastic Compute Cloud</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::EC2::Instance</td>
<td>cmdb_ci_aws_vm_instance</td>
<td>AWS Elastic Compute Cloud</td>
<td>AWS Elastic Compute Cloud</td>
</tr>
<tr>
<td>Service Category</td>
<td>Name</td>
<td>CI Class</td>
<td>Pattern</td>
<td>Product</td>
</tr>
<tr>
<td>------------------</td>
<td>------</td>
<td>---------------</td>
<td>--------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::EC2::InstanceType</td>
<td>cmdb_ci_compute_template</td>
<td>AWS Elastic Compute Cloud</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::EC2::InternetGateway</td>
<td>cmdb_ci_internet_gateway</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AWS::EC2::Keypair</td>
<td>cmdb_ci_cloud_key_pair</td>
<td>AWS Key Management Service</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::NatGateway</td>
<td>cmdb_ci_nat_gateway</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::NetworkAclcmdb_ci_network_acl</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::NetworkIntercmbd_ci_nic</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::RouteTablecmdb_ci_route_table</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>AWS::EC2::SecurityGroup</td>
<td>cmdb_ci_compute_security_group</td>
<td>AWS Elastic Compute Cloud</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::Subnet</td>
<td>cmdb_ci_cloud_subnet</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
</tr>
<tr>
<td>Storage</td>
<td>AWS::EC2::Volume</td>
<td>cmdb_ci_storage_volume</td>
<td>AWS Elastic Block Storage</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::VPC</td>
<td>cmdb_ci_cloud_network</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::EC2::VPNGateway</td>
<td>cmdb_ci_virtual_pvt_gateway</td>
<td>AWS Virtual Private Cloud</td>
<td></td>
</tr>
<tr>
<td>Load Balancer</td>
<td>AWS::ElasticLoadBalancing</td>
<td>cmdb_ci_cloud_load_balancer</td>
<td>AWS Elastic Load Balancing</td>
<td></td>
</tr>
<tr>
<td>Load Balancer</td>
<td>AWS::ElasticLoadBalancing</td>
<td>cmdb_ci_cloud_load_balancer</td>
<td>AWS Elastic Load Balancing</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AWS::IAM::Group</td>
<td></td>
<td>AWS Identity and Access Management</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AWS::IAM::Policy</td>
<td></td>
<td>AWS Identity and Access Management</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AWS::IAM::Role</td>
<td></td>
<td>AWS Identity and Access Management</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AWS::IAM::User</td>
<td></td>
<td>AWS Identity and Access Management</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>AWS::KMS</td>
<td></td>
<td>AWS Key Management Service</td>
<td></td>
</tr>
</tbody>
</table>
## Service categories discovered in Azure
The Discovery process discovers Azure services.

### Services categories discovered in Azure

<table>
<thead>
<tr>
<th>Service Category</th>
<th>Name</th>
<th>CI Class</th>
<th>Pattern</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>AWS::RDS::DBEvent</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Relational Database Service</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>AWS::RDS::DBInstance</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Relational Database Service</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>AWS::RDS::DBSecurityGroup</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Relational Database Service</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>AWS::RDS::DBSnapshot</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Relational Database Service</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>AWS::RDS::DBSubnet</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Relational Database Service</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>AWS::Route53::HostedZone</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Route 53</td>
<td></td>
</tr>
<tr>
<td>Storage</td>
<td>AWS::S3::Bucket</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Simple Storage Service</td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td>AWS::SNS</td>
<td>cmdb_ci_cloud_database</td>
<td>AWS Simple Notification Service</td>
<td></td>
</tr>
<tr>
<td>Azure Datacenter</td>
<td>cmdb_ci_azure_datacenter</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft Data Center</td>
<td></td>
</tr>
<tr>
<td>Azure Subscription</td>
<td>cmdb_ci_azure_subscription</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft Subscription Service</td>
<td></td>
</tr>
<tr>
<td>Cloud Resource</td>
<td>cmdb_ci_cmp_resource</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft Cloud Resource</td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>Microsoft</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft Application Service</td>
<td></td>
</tr>
<tr>
<td>Services</td>
<td>(Unknown Resource Type)</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft Application Service</td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>Microsoft.ApiManagement/service</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.ApiManagement Service</td>
<td></td>
</tr>
<tr>
<td>Services</td>
<td>Microsoft.Cache</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Cache</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Compute/ availabilitySets</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Compute</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Cache</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Cache</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Cache</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Cache</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Cache</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Cache</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Cache</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Cache</td>
<td></td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Cache</td>
<td>cmdb_ci_azure_datacenter</td>
<td>Microsoft.Cache</td>
<td></td>
</tr>
<tr>
<td>Service Category</td>
<td>Name</td>
<td>CI Class</td>
<td>Pattern</td>
<td>Product</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------</td>
<td>----------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>Tools</td>
<td>Microsoft.Insights/components</td>
<td></td>
<td></td>
<td>Microsoft.Insights</td>
</tr>
<tr>
<td>Network</td>
<td>Microsoft.Network/applicationGateways</td>
<td></td>
<td></td>
<td>Microsoft.Network</td>
</tr>
<tr>
<td>Network</td>
<td>Microsoft.Network/networkInterfaces cmdb_ci_nic</td>
<td></td>
<td></td>
<td>Microsoft.Network</td>
</tr>
<tr>
<td>Network</td>
<td>Microsoft.Network/routeTables cmdb_ci_route_table</td>
<td></td>
<td></td>
<td>Microsoft.Network</td>
</tr>
<tr>
<td>Network</td>
<td>Microsoft.Network/subnets cmdb_ci_cloud_subnet</td>
<td></td>
<td></td>
<td>Microsoft.Network</td>
</tr>
<tr>
<td>Network</td>
<td>Microsoft.Network/virtualNetworks cmdb_ci_network</td>
<td></td>
<td></td>
<td>Microsoft.Network</td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Resources/cmdb_ci_azure_deployment deployments</td>
<td></td>
<td></td>
<td>Microsoft.Compute</td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.Resources/cmdb_ci_resource_group resourceGroups</td>
<td></td>
<td></td>
<td>Microsoft.Compute</td>
</tr>
<tr>
<td>Tools</td>
<td>Microsoft.ServiceBus</td>
<td></td>
<td></td>
<td>Microsoft.ServiceBus</td>
</tr>
<tr>
<td>Compute</td>
<td>Microsoft.ServiceFabric/clusters</td>
<td></td>
<td></td>
<td>Microsoft.Compute</td>
</tr>
<tr>
<td>Database</td>
<td>Microsoft.Sql/servers cmdb_ci_cloud_database</td>
<td></td>
<td></td>
<td>Microsoft.Sql</td>
</tr>
<tr>
<td>Database</td>
<td>Microsoft.Sql/servers/databases</td>
<td></td>
<td></td>
<td>Microsoft.Sql</td>
</tr>
<tr>
<td>Storage</td>
<td>Microsoft.Storage/blobs cmdb_ci_storage_volume</td>
<td></td>
<td></td>
<td>Microsoft.Storage</td>
</tr>
<tr>
<td>Storage</td>
<td>Microsoft.Storage/storageAccounts cmdb_ci_cloud_storage_account</td>
<td></td>
<td></td>
<td>Microsoft.Storage</td>
</tr>
<tr>
<td>Application</td>
<td>Microsoft.Web/serverFarms</td>
<td></td>
<td></td>
<td>Microsoft.Web</td>
</tr>
<tr>
<td>Application</td>
<td>Microsoft.Web/sites cmdb_ci_cloud_webserver</td>
<td></td>
<td></td>
<td>Microsoft.Web</td>
</tr>
</tbody>
</table>

**Cloud Admin Portal**

The Cloud Admin Portal is a role-based portal. You can manage, design, govern, operate, and analyze all your cloud resources from a unified base.

Depending on the role you are logged in with, the Cloud Admin Portal displays different metrics. As an example, if you are logged in to the Cloud Management application as a Cloud Service Designer, the Cloud Admin Portal displays only the **Design** metrics.
Log in to the portal with the following Cloud Management roles to view or manage the associated metrics:

<table>
<thead>
<tr>
<th>Descriptive name and role name</th>
<th>Access to the metrics in Cloud Admin Portal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root administrator</td>
<td>All the metrics.</td>
</tr>
<tr>
<td>[sn_cmp.cmp_root_admin]</td>
<td>• Manage</td>
</tr>
<tr>
<td></td>
<td>• Design</td>
</tr>
<tr>
<td></td>
<td>• Govern</td>
</tr>
<tr>
<td></td>
<td>• Operate</td>
</tr>
<tr>
<td></td>
<td>• Analyze</td>
</tr>
<tr>
<td>Cloud administrator</td>
<td>All the metrics.</td>
</tr>
<tr>
<td>[sn_cmp.cloud_admin]</td>
<td>• Manage</td>
</tr>
<tr>
<td></td>
<td>• Design</td>
</tr>
<tr>
<td></td>
<td>• Govern</td>
</tr>
<tr>
<td></td>
<td>• Operate</td>
</tr>
<tr>
<td></td>
<td>• Analyze</td>
</tr>
<tr>
<td>System administrator</td>
<td>All the metrics.</td>
</tr>
<tr>
<td>[admin]</td>
<td>• Manage</td>
</tr>
<tr>
<td></td>
<td>• Design</td>
</tr>
<tr>
<td></td>
<td>• Govern</td>
</tr>
<tr>
<td></td>
<td>• Operate</td>
</tr>
<tr>
<td></td>
<td>• Analyze</td>
</tr>
<tr>
<td>Governor</td>
<td>• Govern</td>
</tr>
<tr>
<td>[sn_cmp.cloud_governor]</td>
<td>• Analyze</td>
</tr>
<tr>
<td>Service Designer</td>
<td>• Design</td>
</tr>
<tr>
<td>[sn_cmp.cloud_service_designer]</td>
<td></td>
</tr>
<tr>
<td>Operator</td>
<td>• Operate</td>
</tr>
</tbody>
</table>

Only the Root administrator, the Cloud administrator, and the System administrator roles can view and manage all the metrics in the Cloud Admin Portal.

Click **Guided Setup** to configure the Cloud Management application on your ServiceNow instance. You are guided through completing various configuration tasks. The percentage of the tasks that you have completed is displayed for you to monitor your progress.

Click **Get Help** to ask a question, open an incident, go to the product documentation or the Knowledge base website.

**Resource profiles**

Resource profiles are cloud provider-agnostic definitions that specify the allowed attribute values for a resource. Resource profiles enable you to control the choices that the user sees when requesting a cloud resource. As a result, you do not need to define a unique blueprint for each variation of the resource.
Note: In addition to the cloud-agnostic profiles, you can add provider public images to the Cloud Management catalog. See Add an AWS public image to Cloud Management and Add an Azure public image to Cloud Management.

Example: Compute profile

You might create a compute profile named Micro and map it to a "micro" hardware type, with specified values for memory and CPU. Later, when a user selects a compute resource while requesting a virtual server, Micro appears as an option in the list of available compute resources. The user does not need to specify values for memory and CPU — the values come from the profile.

Resource profile mappings

Resource profiles create mapping associations between the following items:

- A cloud account.
- A logical datacenter in the cloud account.
- A specific resource type in the CMDB that provides the attributes.
- Optional: A pricing value that appears to users when they request a resource that uses the resource profile.

By default for AWS and Azure, Cloud Management maps profiles to templates after Discovery runs. For other providers, such as Google cloud, you must manually associate the profile with the correct template and datacenter.
## Resource types

Each resource profile has associated attributes. The attributes depend on the resource types that are populated after you run Discovery on a cloud account.

<table>
<thead>
<tr>
<th>Resource profile type</th>
<th>Description and attributes</th>
<th>Resource type and template</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application profile</td>
<td>An application profile specifies application software to install on newly-provisioned resources. Users can select applications when they request a stack. Use application profiles when you integrate with configuration management (continuous delivery) providers such as Ansible playbooks. Attributes: Applications running on virtual machines.</td>
<td>Application Template (sn_cmp_application_template)</td>
</tr>
<tr>
<td>Compute profile</td>
<td>A compute profile specifies the hardware to use for newly-provisioned virtual machines. Attributes: The size of computing resources, including the virtual CPUs, memory, and local storage.</td>
<td>Hardware Type (cmdb_ci_compute_template)</td>
</tr>
<tr>
<td>Compute Security Group profile</td>
<td>A compute security group profile applies specified security rules to newly-provisioned resources. You map a compute security group profile to the following items: a cloud account, a datacenter, a Compute Security Group template, and security rules for the template. Attributes: Firewall rules, such as enabling HTTP and HTTPS.</td>
<td>Compute Security Group Template (cmdb_ci_security_grp_template)</td>
</tr>
<tr>
<td>OS profile</td>
<td>An OS profile installs a specified image on a newly-provisioned virtual machine. You map an OS profile to a cloud account, a location (datacenter), an image template, and a cloud script. Attributes: Operating system images, including the OS type and version, the root device type, and the image source.</td>
<td>Image (cmdb_ci_os_template)</td>
</tr>
<tr>
<td>Schedule profile</td>
<td>You map a schedule profile to an instance schedule. The schedule profile applies to all newly-provisioned resources that use the profile. For example, a schedule profile can specify the days of the week and times of day when a stack should start and stop. Attributes: Schedule attributes such as when a stack should be started, stopped, or deprovisioned.</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Create an application profile

An application profile specifies application software to install on newly-provisioned resources. Users can select applications when they request a stack. Use application profiles when you integrate with configuration management (continuous delivery) providers such as Ansible playbooks.

- Role required: sn_cmp.cloud_admin
- You must have a cloud account with datacenters. You must run Discovery on the service accounts to populate the datacenters.

This example shows an application profile mapping for a Tomcat server on Ansible.
1. In the Cloud Admin Portal, navigate to Manage > Resource Profiles.
2. In the Profiles list, select Application Profile and then click New.
3. Enter a unique and descriptive Name and Description for the profile and then click Submit. The profile is created.

4. Map the profile to a template.
   a) In the list, click the profile that you created.
   b) In the Application Profile Mappings related list, click New, fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workload Config Provider</td>
<td>Select a configuration management provider to associate with the profile.</td>
</tr>
</tbody>
</table>
| Application Template          | Select a template that the profile should be mapped to. If you ran Discovery on the Configuration Provider, the templates are already populated. The resource type associated with an application profile is sn_cmp_application_template. Click the reference icon (i) to view the details of the template. To create a new template:
   1. Click the reference icon (i) to open the Application Templates list.
   2. Click New and then fill in the Application Template form:
      - Name: Enter a descriptive name for the template.
      - Template ID: Enter an ID to use for the template.
      - Config runlist provider: Click the list, select a table from the Table name list, and then select records for the table from the Document list.
      - Config runlist: Specify a configuration runlist.
      - Provider instance: Select a provider instance. |

5. Create as many mappings as needed.

Create a compute profile

A compute profile specifies the hardware to use for newly-provisioned virtual machines. A compute profile maps to a cloud account, a datacenter, and a hardware template.

   - Role required: sn_cmp.cloud_admin
   - You must have a cloud account with datacenters. You must run Discovery on the service accounts to populate the datacenters.

1. In the Cloud Admin Portal, navigate to Manage > Resource Profiles.
2. In the Profiles list, select Compute Profile and then click New.
3. Enter a unique and descriptive **Name** and **Description** for the profile and then click **Submit**. The profile is created.

4. Click the resource profile that you created.

5. Map the profile to a template.
   a) In the list, click the profile that you created.
   b) In the **Profile Mappings** related list, click **New**, fill in the form, and then click **Submit**.

Example compute profile

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Account</td>
<td>Select a cloud account that the profile is valid for.</td>
</tr>
<tr>
<td>Location</td>
<td>Select the datacenter that the profile is valid for.</td>
</tr>
<tr>
<td>Hardware Template (cmdb_ci_compute_template)</td>
<td>Select the hardware type that the profile should be mapped to.</td>
</tr>
<tr>
<td></td>
<td>Click the reference icon (i) to view the details of the template.</td>
</tr>
</tbody>
</table>

Create a compute security group profile

A compute security group profile applies specified security rules to newly-provisioned resources. You map a compute security group profile to the following items: a cloud account, a datacenter, a Compute Security Group template, and security rules for the template.
Role required: sn_cmp.cloud_admin
You must have a cloud account with datacenters. You must run Discovery on the service accounts to populate the datacenters.

1. In the Cloud Admin Portal, navigate to Manage > Resource Profiles.
2. In the Profiles list, select Compute Security Group Profile and then click New.
3. Enter a unique and descriptive Name and Description for the profile and then click Submit. The profile is created.
4. Map the profile to a template,
   a) In the list, click the profile that you created.
   b) In the Compute Security Group Profile Mappings related list, click New, fill in the form, and then click Submit.

```
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Account</td>
<td>Select a cloud account for the profile.</td>
</tr>
<tr>
<td>Location</td>
<td>Select the datacenter that belongs to the cloud account.</td>
</tr>
<tr>
<td>Compute Security Group Template</td>
<td>Select or create a template that the profile should be mapped to.</td>
</tr>
<tr>
<td></td>
<td>To create a new template, click the list icon and then click New. Enter a name and a template ID and click Submit.</td>
</tr>
<tr>
<td></td>
<td>Click the reference icon (i) to view the details of the template.</td>
</tr>
</tbody>
</table>
```

5. Add rules to the template. You can create more than one rule per template.
Create an OS profile

An OS profile installs a specified image on a newly-provisioned virtual machine. You map an OS profile to a cloud account, a location (datacenter), an image template, and a cloud script. OS profiles are provider-agnostic and you can use the same profile for multiple cloud accounts.

- Role required: sn_cmp.cloud_admin
- You must have a cloud account with datacenters. You must run Discovery on the service accounts to populate the datacenters.

Create an OS profile

An OS profile installs a specified image on a newly-provisioned virtual machine. You map an OS profile to a cloud account, a location (datacenter), an image template, and a cloud script. OS profiles are provider-agnostic and you can use the same profile for multiple cloud accounts.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIDR Range</td>
<td>Specify a CIDR range. For example, 10.0.0.0/24.</td>
</tr>
<tr>
<td>IP Protocol</td>
<td>Select an IP protocol. In Azure, the protocols supported are TDP and UDP. ICMP is not supported.</td>
</tr>
<tr>
<td>From Port</td>
<td>Specify the source port number.</td>
</tr>
<tr>
<td>Is Outbound</td>
<td>Enter 0 for inbound and 1 for outbound.</td>
</tr>
<tr>
<td>Name</td>
<td>Specify a name for the rule.</td>
</tr>
<tr>
<td>Network Type</td>
<td>Select the type of network.</td>
</tr>
<tr>
<td>To Port</td>
<td>Specify the destination port number.</td>
</tr>
</tbody>
</table>

Note: Profile mappings that specify more details run first. For example, a mapping that specifies a blueprint, OS profile, and resource alias takes precedence over a mapping that specifies only an OS profile.

1. In the Cloud Admin Portal, navigate to Manage > Resource Profiles.
2. In the Profiles list, select OS Profile and then click New.
3. Enter a unique and descriptive Name and Description for the profile and then click Submit. The profile is created.
4. Map the profile to a template.
   a) In the list, click the profile that you created.
   b) In the OS Profile Mappings related list, click New, fill in the form, and then click Submit.
5. Map the profile to a cloud script. In this procedure, you specify an existing script. See [Create cloud initialization script templates and a script](#).
a) On the Cloud Script OS Profile Mappings related list, click **New** and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blueprint</td>
<td>(Optional) Select a blueprint to limit the script to a specific resource block used in the blueprint.</td>
</tr>
<tr>
<td>Cloud script</td>
<td>Select a cloud script to map the OS profile to.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box if the cloud script should be run after the virtual machine is provisioned.</td>
</tr>
<tr>
<td>OS profile</td>
<td>If you specify a blueprint, the cloud script is run when the blueprint is provisioned.</td>
</tr>
<tr>
<td>Resource Alias</td>
<td>(Optional) If you specify a resource alias for the blueprint, then the cloud script is executed when the blueprint with the specified resource alias is provisioned.</td>
</tr>
</tbody>
</table>

b) Right-click the form header and select **Save**. The **Name** attribute `[scriptName]` appears in the OS Profile Mapping Overrides list.

Optional: In the OS Profile Mapping Overrides list, enter an attribute name and value to use when the resource is provisioned.

Create a schedule profile

You map a schedule profile to an instance schedule. The schedule profile applies to all newly-provisioned resources that use the profile. For example, a schedule profile can specify the days of the week and times of day when a stack should start and stop.

- **Role required:** `sn_cmp.cloud_admin`
- You must have a cloud account with datacenters. You must run Discovery on the service accounts to populate the datacenters.

1. In the Cloud Admin Portal, navigate to **Manage > Resource Profiles**.
2. In the **Profiles** list, select **Schedule Profile**.
3. Click **New**, enter a unique and descriptive **Name** and **Description**, fill in the form, and then click **Submit**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule</td>
<td>Select an existing schedule or define a custom schedule. A schedule is a rule that includes or excludes specified time periods for actions.</td>
</tr>
<tr>
<td>Multiplier</td>
<td>Auto-populated value between 0 and 1: The fraction of time that a stack with this schedule is expected to be powered on.</td>
</tr>
</tbody>
</table>
Add credentials to an image template
When you add credentials to an image template, the credentials are inherited by all VMs that are provisioned using the template.

- Role required: cloud_admin
- A set of basic authentication credentials must exist.

1. Perform one of the following actions:
   - If an OS profile is mapped to the virtual machine template:
     1. Open the OS profile (Manage > Resource Profiles).
     2. In the OS Profile Mappings related list, click the Name of the template in the Image Template column.
   - For VMware templates, navigate to Configuration > VMware > Virtual Machine Templates.

2. On the Image form, select the basic authentication Credentials.
3. Click Update.

Add an AWS public image to Cloud Management
You can make any AWS image selectable in an OS profile.

Role required: sn_cmp.cloud_admin

1. In the AWS console, select the EC2 service and then click IMAGES > AMIs.

2. Filter the list of images and then take note of the AMI Name and AMI ID values for the image of interest.
3. On the instance, enter cmdb_ci_os_template.list in the Filter navigator text box to open the Images (cmdb_ci_os_template) table and then click New.
4. On the Image form, enter the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object ID</td>
<td>AMI ID value from AWS.</td>
</tr>
<tr>
<td>Name</td>
<td>Meaningful name that is based on the AMI Name value from AWS.</td>
</tr>
</tbody>
</table>
5. On the **Credentials** field, click the magnifying glass and then click **New** on the Credentials page.

6. In the list, click **Basic Auth Credentials**.

7. On the Basic Auth Credentials form, enter a meaningful **Name** for the credential.

8. Enter a **User name** and **Password** to use for instances that are provisioned from this image.

   **Note:** If the name `ec2-user` or `root` does not work, check with your AMI provider.

### Username guidelines

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Linux</td>
<td>ec2-user</td>
</tr>
<tr>
<td>RHEL</td>
<td>ec2-user or root</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>ubuntu or root</td>
</tr>
<tr>
<td>Centos</td>
<td>centos</td>
</tr>
<tr>
<td>Fedora</td>
<td>ec2-user</td>
</tr>
<tr>
<td>SUSE</td>
<td>ec2-user or root</td>
</tr>
</tbody>
</table>

9. Click **Submit** to close the Basic Auth Credentials form.

10. Right-click in the header and select **Save** to save the current configuration of the new image record.

   In the Related Items list, you can see that the image is associated with the specified datacenter.

11. Click **Update** to make the public image selectable in an OS profile.

   This step also creates a relationship between the public image and the logical datacenter (LDC) in the CMDB.

---

### Add an Azure public image to Cloud Management

You can make any Azure image selectable in an OS profile.

**Role required:** sn_cmp.cloud_admin

1. In the Azure command-line interface, make note of the values of the **Offer**, **Publisher**, **Sku**, and **Version**.

   On the command line, you can search by offer or by publisher or by both, as show in the example. Each line item is an image of a particular version of CentOS from OpenLogic.
2. On the instance, enter `cmdb_ci_os_template.list` in the Filter navigator text box to open the Images (cmdb_ci_os_template) table and then click New.
3. Configure the form layout to show Offer, Serial number, Version, and Vendor.
4. On the Image form, enter the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object ID</td>
<td>Enter <code>/publisher/offer/sku/version</code>. You can get these values from the Azure CLI. As an example, <code>/OpenLogic/CentOS/6.5/6.5.201501</code>.</td>
</tr>
<tr>
<td>Name</td>
<td>Meaningful name that represents the version of the OS. CentOS 6.5 in the example.</td>
</tr>
<tr>
<td>Guest OS</td>
<td>Operating system.</td>
</tr>
<tr>
<td>Offer</td>
<td>Offer value from Azure.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>Sku value from Azure.</td>
</tr>
<tr>
<td>Version</td>
<td>Version value from Azure.</td>
</tr>
<tr>
<td>Vendor</td>
<td>1. Click the magnifying glass and then click New.</td>
</tr>
<tr>
<td></td>
<td>2. In the Name field, enter the Publisher value from Azure.</td>
</tr>
<tr>
<td></td>
<td>3. Click Submit.</td>
</tr>
</tbody>
</table>

5. On the Credentials field, click the magnifying glass and then click New on the Credentials page.
6. In the list, click Basic Auth Credentials.
7. On the Basic Auth Credentials form, enter a meaningful Name for the credential.
8. Enter a User name and Password to use for instances that are provisioned from this image.
9. Click Submit to close the Basic Auth Credentials form.
10. Right-click in the header and select Save to save the current configuration of the new image record.
    In the Related Items list, you can see that the image is associated with the specified datacenter.
11. Click Update to make the public image selectable in an OS profile.
    This step also creates a relationship between the public image and the logical datacenter (LDC) in the CMDB.
**IPAM integration**
If you use an IP address management (IPAM) tool such as Infoblox, you can manage cloud IP addresses, networks, and subnets in your cloud catalog offerings.

**Infoblox integration**

By default, Cloud Management supports Infoblox integration. You can create Infoblox credentials and then set up an Infoblox instance record. See [Register an Infoblox server](#) for instructions.

**Subnets**

If you have subnets in your AWS and Azure clouds, Cloud Management finds them when Discovery runs on a service or cloud account.

**Note:** If you have networks and subnets in a VMware vSphere platform, Discovery does not create subnet records in the CMDB and cannot map the subnet to the appropriate network. You must create the records manually.

**Register an Infoblox server**

To integrate with Infoblox, you must create an Infoblox record and associate it with the appropriate credentials to register your Infoblox server with the instance.

- Role required: sn_cmp.cloud_admin
- Create [Infoblox credentials](#).

1. In the Cloud Admin Portal, navigate to Manage > Networks & IPAM > IPAM Providers > Infoblox.
2. Click **New**, enter a unique and descriptive **Name**, fill in the form, and then click **Submit**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td>Enter the URL of the Infoblox server.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select the Infoblox credential.</td>
</tr>
</tbody>
</table>

**Create an IP pool**

Create an IP pool to associate a cloud subnet with an IPAM provider instance.

- Role required: sn_cmp.cloud_admin
- Create [Infoblox credentials](#) and [register an Infoblox server](#).
- Run Discovery on the cloud account to find subnets or [manually create cloud subnets](#).

1. In the Cloud Admin Portal, navigate to Manage > Networks & IPAM.
2. On the **IP Pools** tab, click **New**, enter a unique and descriptive **Name**, and then fill in the form.
Field | Description
--- | ---
IPAM Provider | Select an IPAM provider.
Provider Instance | Select an instance for the provider.
Provider Version | Enter a version for the provider.
Subnet | Select a subnet. For AWS or Azure, select a discovered subnet that has been configured in AWS or Azure. For VMware, copy the CIDR range from the IPAM tool and paste the CIDR range in this field. See [Create a cloud subnet](#).
Mode | Select a mode based on the provider that you are creating this IP pool for. The model varies based on the provider.

3. Click **Submit**.

The related subnet appears in the **Subnet** related list at the bottom of the form.

Create a cloud subnet

If you have subnets in your VMware vSphere account, create a cloud subnet record in the instance.

Role required: sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to **Manage > Networks & IPAM**.
2. On the **IP Pools** tab, click **New** to create a new record.
3. In the **Subnet** field, click the lookup icon.
   The Cloud Subnets list appears.
4. Click **New**.
5. Fill out the form fields (see table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive name.</td>
</tr>
<tr>
<td>Status</td>
<td>Leave the status value as <strong>Installed</strong>.</td>
</tr>
<tr>
<td>CIDR/Gateway/Subnet Mask</td>
<td>Enter the CIDR associated with the subnet, the IP address of the gateway, and the subnet mask. For VMware, the CIDR, Gateway, and Subnet Mask fields are mandatory.</td>
</tr>
<tr>
<td>Primary and Secondary DNS</td>
<td>Enter the DNS values. For VMware, the Primary DNS field is mandatory.</td>
</tr>
</tbody>
</table>

6. Click **Submit**.

Create a relationship between a VMware network and subnet  
If you discover VMware networks and subnets, you must manually create a relationship between the two.

Role required: sn_cmp.cloud_admin

If no relationship exists and a user must select a virtual network and subnet in the Cloud User Portal, the user can encounter an error.

1. In the Cloud Admin Portal, navigate to **Manage > Service Accounts**.
2. Open the datacenter that contains the VMware cloud network.

   ![Note: VMware cloud networks are saved in the VMware vCenter Network (cmdb_ci_vcenter_network) table. AWS and Azure cloud networks are saved in the Cloud Networks (cmdb_ci_network) table.](image)

   3. Under **Related Items**, click the plus icon to add a new relationship.
   4. Make sure the **Use suggested relationships** check box is selected, and then select the **Contains (Parent)** relationship.
   5. Under **Filter**, create the filter necessary to find the subnet.
   6. Under **Configuration Items**, select the check box next to the subnet.
   7. In the **Relationships** section, click the plus icon.

   The relationship is added to the list.

8. Click **Save and Exit**.

Reserve IP addresses for VMware vSphere virtual machines in InfoBlox

Create a policy to reserve IP addresses for VMware vSphere virtual machines in Infoblox, at the time of provisioning the virtual machines.

Role required: sn_cmp.cloud_admin.

You can create a policy that is invoked when a VMware virtual machine is provisioned to get an IP address from Infoblox. Before the virtual machine is provisioned, the instance makes a call to Infoblox that creates a host record with the IP address.

1. In the Cloud Admin Portal, navigate to **Govern > Policies**.
2. Click **New** and then fill in the form.
Register IP addresses for AWS and Azure virtual machines in Infoblox

Create a policy to register IP addresses for AWS and Azure virtual machines in Infoblox, once these virtual machines are provisioned.

Role required: sn_cmp.cloud_admin.

When AWS and Azure virtual machines are provisioned, IP addresses are automatically allocated to them. You can create a policy and the policy is invoked, once these virtual machines are provisioned, to get their IP addresses and register these IP addresses in Infoblox.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Click New and then fill in the form.
   
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Name</td>
<td>Enter a descriptive name.</td>
</tr>
<tr>
<td>Policy Trigger</td>
<td>Select on Resource operation.</td>
</tr>
<tr>
<td>Resource Block</td>
<td>Select ResourceBlock Virtual Server.</td>
</tr>
<tr>
<td>Operation</td>
<td>Select Virtual Server interface provision.</td>
</tr>
<tr>
<td>Moment</td>
<td>Select Pre Operation.</td>
</tr>
</tbody>
</table>

3. Click Submit to publish the policy.
4. Create a policy rule for the policy. Specify two conditions in the condition builder for locating the vSphere virtual machine. Select Request from > starts with and enter vSphere.
5. Click Update.
6. In the Policy Rules Action related list, click New to create a policy action.
7. Select IP Address Management in the Create Policy Rule Action dialog box. Enter a name in the Action Name field, select Infoblox in the Provider Type list, and select Reserve IP address in the IPAM Method Name list.
8. Click Submit.
9. In the Policy Rule Action Attributes section, enter the domain name of your network in the DNNSuffix field. As an example, corp.servicenow.com. Values for all the other properties are already populated.
10. Click Update and republish the policy.
8. Click Submit.
9. In the Policy Rule Action Attributes section, enter the domain name of your company’s network in the DNSSuffix field. As an example, corp.servicenow.com. Values for all the other properties are already populated.
10. Click Update and republish the policy.

Delete a host record in Infoblox
When a virtual machine is de-provisioned, delete the host record from Infoblox. Deleting the host record enables that IP address to be available and be reused for other virtual machines.

Role required: sn_cmp.cloud_admin.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Click New.
3. Fill out the fields on the Policy form (see table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Name</td>
<td>Enter a descriptive name.</td>
</tr>
<tr>
<td>Policy Trigger</td>
<td>Select on Resource operation.</td>
</tr>
<tr>
<td>Resource Block</td>
<td>Select ResourceBlock Virtual Server.</td>
</tr>
<tr>
<td>Operation</td>
<td>Select Virtual Server interface Deprovision.</td>
</tr>
<tr>
<td>Moment</td>
<td>Select Post Operation.</td>
</tr>
</tbody>
</table>

4. Click Submit to publish the policy.
5. Create a policy rule for the policy and click Update.
6. In the Policy Rules Action related list, click New to create a policy action.
7. Select IP Address Management in the Create Policy Rule Action dialog box. Enter a name in the Action Name field, select Infoblox in the Provider Type list, and select Release IP address in the IPAM Method Name list.
8. Click Submit.
9. In the Policy Rule Action Attributes section, enter the domain name of your company network in the DNSSuffix field. As an example, corp.servicenow.com. Values for all the other properties are already populated.
10. Click Update and republish the policy.

Support for continuous delivery (configuration management)
The Cloud Management application supports integration with continuous delivery solutions (also known as configuration management). Ansible is supported as the default config management provider.

The Cloud Management application supports configuration management providers in two ways:

- **Application profile based approach**: Create an application profile for an entity and in the order catalog form, select that profile to deploy that entity. Each profile has a config installable. You can create multiple application profiles- as an example, one for PostgreSQL and one for Apache. Each profile can be mapped to a configuration management provider. In the Cloud User Portal, based on the profile you select, a configuration grid appears populated with attributes and values for that specific configuration management provider. You can modify any values in the order form and provision the entity.
- **Resource block based approach**: Use a resource block, like Apache2, in a blueprint to represent an entity that a configuration management provider manages. The same resource block can be used to support more than one entity of the same kind. As an example, an Apache2 resource block can be used to support multiple Apache2 servers. In the order catalog form, select a provider type and then select a provider. Based on the specific provider,
configuration management attributes and values appear. Once you provision the entity, you can save the configuration of the resource block and store it in the CMDB for future use.

What to do

<table>
<thead>
<tr>
<th>Step</th>
<th>Goal</th>
<th>See these topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Create credentials for Ansible.</td>
<td>Enable Cloud Management to access the configuration provider with the necessary credentials.</td>
<td>- Configure Ansible Tower user name and password</td>
</tr>
<tr>
<td>2. Create the workload provider type, and then create the provider.</td>
<td>Configure Cloud Management to work with the configuration management provider of your choice. Then run discovery on the provider, using the credentials you provide, to find the resources that the provider already owns.</td>
<td>- Create workload provider type and Create a configuration management provider and run Discovery</td>
</tr>
<tr>
<td>3. Create an application profile with a mapping</td>
<td>Create the profile that defines the application that the configuration management provider manages. The profile is based on the discovered inventories in the provider.</td>
<td>- Create an application profile</td>
</tr>
<tr>
<td>4. Create a blueprint</td>
<td>Create a blueprint with BootstrapNode, Register Node, and ExecuteConfigPackages operations on the virtual resource that the configuration provider manages. You can also customize the input parameters on the form to allow the user to select important inputs, like the application profile template, organization, and credential ID.</td>
<td>- Create a blueprint for configuration management provider integration and Create a blueprint for Ansible Tower integration</td>
</tr>
<tr>
<td>5. Provision a resource from the Cloud User Portal</td>
<td>The resource should provision, with the stack status indicating the BootstrapNode, Register Node, and ExecuteConfigPackages steps.</td>
<td></td>
</tr>
</tbody>
</table>

Configure Ansible Tower user name and password

Configure Ansible Tower credentials to access the Ansible Automation Engine through your instance.

Role required: sn_cmp.cloud_service_designer

1. Navigate to Orchestration > Credentials, and then click New.
2. Click Ansible Tower Credentials.
3. Fill out the form fields (see table):

To integrate Cloud Management with the Ansible configuration management account, you need only to configure the user name and password for the administrator account in Ansible.
Form fields for Ansible Tower credentials

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive name.</td>
</tr>
<tr>
<td>User name/Password</td>
<td>Enter the authentication credentials for the Ansible Tower user with administrator rights.</td>
</tr>
</tbody>
</table>

Note: You do not need to configure the other fields.

Create workload provider type
Create a workload provider type for each new configuration management provider. This information appears in the order catalog form as management attributes that your users can select when provisioning a virtual resource through a configuration management provider.

Role required: cloud_admin

1. In the Cloud Admin Portal, navigate to Manage > ConfigManagement.
2. Click Workload Config Provider Types, and then click New.
3. Fill in the form fields (as shown in the table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the workload provider type.</td>
</tr>
<tr>
<td>Product Type</td>
<td>Select a product type from the list.</td>
</tr>
<tr>
<td>Config CI</td>
<td>Select the configuration table created for the provider that displays the discovered resources from the list.</td>
</tr>
<tr>
<td>Credential Resolver</td>
<td>Select a credential resolver from the list.</td>
</tr>
<tr>
<td>Server Type</td>
<td>Select the type of server for the provider like Opensourced, Enterprise, etc.</td>
</tr>
<tr>
<td>Version</td>
<td>Enter the version of the provider.</td>
</tr>
<tr>
<td>Credential Type</td>
<td>Select the table where the credentials are stored for this provider.</td>
</tr>
</tbody>
</table>
4. Add properties for the workload type in the **Workload Provider Properties** section. In addition to the existing properties, you can add more properties. Workload provider properties are displayed in the order catalog form (in the Cloud User Portal) as management attributes.

For example, for an Ansible provider type, **Inventory** and **Hostgroup** are required. The values for these properties come from Resource Pools.
5. Click **Submit**.

Create a configuration management provider and run Discovery

Create a configuration management provider such as Chef or Ansible, and then run Discovery on the provider to find its resources.

- Role required: cloud_admin
- For Chef, you need a Chef server and **Chef credentials**.
- For Ansible, you need an Ansible server and **Ansible credentials**.

1. In the Cloud Admin Portal, navigate to **Manage > Config Management**.
2. Click **New**.
3. Fill out the form fields (see table). Obtain most of this information from your Chef server setup.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a descriptive name.</td>
</tr>
<tr>
<td>Organization</td>
<td>Enter the Chef or Ansible Tower organization for access control. You can obtain this information from the configuration management provider console.</td>
</tr>
<tr>
<td>URL</td>
<td>Enter the full URL of your Chef server or Ansible Tower, including the https protocol (for Ansible) and the port number (for Chef).</td>
</tr>
<tr>
<td>Provider</td>
<td>Select <strong>Chef Server 12</strong> or <strong>Ansible Tower</strong>. If you are adding a configuration management provider, create provider type information for that provider. Once you create the provider type information, the provider type appears in the <strong>Provider</strong> list.</td>
</tr>
<tr>
<td>Server Type</td>
<td>Select a server type: For Chef Server 12, select one of these types: Hosted, Private, Enterprise. For Ansible Tower, select one of these types: Opensourced, Licensed.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select Chef credentials.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.
5. On the Config Providers page, click the provider tile.
6. To find the resources in the config provider, click **Discover Now**.

   **Note:** If you provision or otherwise change resources through the configuration management provider interface, you must manually run Discovery again through this configuration management provider form. You cannot create a scheduled discovery for configuration management providers.

The discovered resources appear under **Entities**.

The discovered resources for Chef are:

- Chef Server Cookbook
- Chef Server Node

The discovered resources for Ansible Tower are:

- **Ansible Inventory** categorizes the applications and virtual resources.
- **Cfg Installable** holds the actual components that run when the applications install. For Ansible, the items in this list are *playbooks*, which are also referred to as *runlists* in this form.
Example Ansible Inventory
Ansible config installables (playbooks)

7. Click **Chef Server Cookbook** or **Ansible Inventory** to explore the contents.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
After a user provisions a resource from the blueprint, the Stack Status indicates how the system runs through the Create node, Bootstrap, and ExecuteConfigPackage steps. You can obtain the IP address of a virtual machine in the User Portal by going to Stacks > {category} and selecting the new virtual machine. Open the configuration management provider server to see the newly provisioned resource on the node the user specified.

Create a blueprint for configuration management provider integration

After you set up your configuration management provider and run Discovery on it, configure resource blocks.

Role required: sn_cmp.cloud_service_designer

For an example of a virtual machine configured to work with Ansible, see Create a blueprint for Ansible Tower integration.

1. In the Cloud Admin Portal, navigate to Design > Blueprints.
2. Click New.
3. Create a deployment model with a container, virtual machine, and a datacenter.
4. Click the **Operations** tab, select an operation, and then click **Add**. The Add Resource Operation dialog box appears.

5. Add the following resource operations for the Virtual Server one at a time:
   - **Bootstrap Node**: prepares the virtual server to communicate with Ansible Tower.
   - **Register Node**: registers the configuration management provider.
   - **ExecuteConfigPackages**: runs the applications or playbooks.

6. Click **Submit** once you have added the resource operations. The operations are added to the graph.
7. Make the following changes to these operations, and then click **Save** after each change:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Make this change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blueprint Container Resource.Provision</td>
<td>Enable the <strong>WorkloadConfigProvider</strong> and <strong>WorkloadConfigProvider Type</strong> parameters. On the Cloud User Portal, users can see these fields to select the configuration management provider.</td>
</tr>
</tbody>
</table>

| Management Attributes | Every operation has management attributes. |

The form parameters, **Virtual_Server_ApplicationProfile**, **Virtual_Server_ConfigurationOverrides**, and **Virtual_Server_ManagementAttributes**, are displayed in the Cloud User Portal while launching a catalog.

8. Go to the Cloud User Portal. Select the workload provider type in the order catalog form. Based on the workload provider type you select, the **Workload Config Provider** field gets loaded.

9. Select an application profile. Based on the profile, the configuration overrides appear. These overrides are the discovered config installable parameters.

10. Publish the catalog item.

Create a blueprint for Ansible Tower integration

For resources that a configuration management provider manages, you must add operations beyond the standard **Provision** operation. This example shows you how to create a blueprint for a virtual server in Ansible.

Role required: **sn_cmp.cloud_service_designer**

1. In the Cloud Admin Portal, navigate to **DesignBlueprints**, and then click **New**.
2. Enter a unique and meaningful **Name** and **Description**
3. Specify the service **Category** for the blueprint.
4. Click **Save**.
5. Add a **Virtual Server** resource block and a datacenter.
6. Complete the following steps to create the containment connections.
7. Modify the operations as follows:
   a) Click the **Provision** operation in the **Operations** tab under the deployment model.
b) Next to Provision, click Add.
c) Add these operations to the virtual server resource in the blueprint:

<table>
<thead>
<tr>
<th>Interface</th>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Server Interface</td>
<td>Bootstrap Node</td>
<td>Prepares the virtual resources to communicate with Ansible Tower.</td>
</tr>
<tr>
<td>Virtual Server Interface</td>
<td>Register Node</td>
<td>Registers with the Ansible Tower server that is configured.</td>
</tr>
<tr>
<td>Virtual Server Interface</td>
<td>ExecuteConfigPackages</td>
<td>Runs the playbooks.</td>
</tr>
</tbody>
</table>

**Add Resource Operation**

- **Resource**: Virtual Server
- **Interface**: Virtual Server Interface
- **Operation**: Bootstrap Node

**Bootstrap Node operation**

This graphic illustrates the operations:
d) Click **Provision** in the Blueprint Container Resource row.

```
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>WorkloadConfigProvider</td>
<td>${parameter.WorkloadConfigProvider}</td>
</tr>
<tr>
<td>WorkloadConfigProviderType</td>
<td>${parameter.WorkloadConfigProviderType}</td>
</tr>
</tbody>
</table>
```

If these parameters do not exist, you can create them by clicking the plus (**Add Attributes**) icon.

f) Click **Save**.
g) Click **Provision** in the Virtual Server row.

h) Make sure that the check box next to the **ComputeFirewallName** parameter is selected.

![ComputeFirewallName](image)

i) Click **Save**.

j) Also on the **Provision** operation for the virtual server, change the value for the **DiscoverAndAttachIP** input attribute to **True**. You do not need to select the check box for this parameter because the user does not need to see it.

k) Click **Save**.

l) Click the **Bootstrap** operation for the virtual server and make sure that the check box next to **ManagementAttributes** is selected. On the Cloud User Portal, users select the inventory and the actual application they want to install from the **Virtual Server Management Attributes** field.

m) Click the **ExecuteConfigPackages** operation for the virtual server and make sure that the check box next to **ConfigurationOverrides** is selected.

On the Cloud User Portal, users can specify values for several attributes of the application, such as the username and password, from the **Virtual Server Configuration Overrides** field.

The settings that can be overridden come from the Ansible Tower template. Make sure that the values in the **EXTRA VARIABLES** field in the Ansible template are saved in JSON so that the MID Server can process the data.
8. Click **Generate Catalog**.
9. Click the **Draft/Published** slider to **Published**.

**Day 2 operations using workflows**

Take advantage of the workflows framework to automate your Day 2 operations. Quickly write a workflow that communicates with a Cloud API or a particular resource. Use SSH, PowerShell, or a similar tool, to access and then extend the workflow capabilities.

Day 1 operations provision resources in a cloud as part of ordering from the ServiceNow catalog or the cloud catalog. These operations result in a stack that contains a list of provisioned resources. Day 2 operations can be carried out on the resources that are part of the stack or the resources discovered by the system.

Previously, to execute Day 2 operations in Cloud Management, you had to interact directly with the Cloud API (CAPI), which frequently meant that you had to write new APIs to support a new operation. With the ability to map Day 2 operations to workflows, you can now more easily use base-system workflow capabilities.

Capabilities include:
- SSH/PowerShell to connect to a VM.
- Use REST end points to call various Cloud Providers.
Rules with workflows

Cloud Management supports using Now Platform rules with workflows. Rules are collections of conditions and actions. If all conditions of a rule evaluate to true, the system performs the actions. If any condition evaluates to false, the system does not perform the actions. Creating rules helps you track activities and more quickly respond to and resolve issues.

Learn more

To learn about workflows in general, see:

- Workflow
- Create a workflow from a table

To learn about how to work with variables in workflows, see:

- Workflow input variables
- Using variables in a workflow

Credential handling

Extend the workflow engine to manage processes and automate things outside of an instance with Orchestration. Use the appropriate credentials required by Orchestration: SSH requires SSH and PowerShell activity elements: SSH requires SSH and PowerShell requires Windows.

The Cloud Management application generates credentials (node credentials) after a VM is provisioned. Node credentials are used to connect to a VM. In addition to node credentials, the system generates SSH/Windows credentials. These credentials are required for Orchestration, which provides the automation for workflows.

Pre-existing VMs

Cloud Management now generates SSH private key credentials and Windows credentials for all active pre-existing VMs provisioned through Cloud Management.

Delete credentials

When you deprovision or delete a VM, all associated credentials are deleted, including the node credential and SSH private key credentials/Windows credentials, as well as the credentials alias tag.

Use the credentials

See Use credentials with SSH workflows for details.

Use credentials with SSH workflows

Add an SSH workflow with a credentials tag.

- Orchestration must be installed.
- Role required: Cloud designer or sn_cmp.cloud_admin

You can use an expression to get the Credential alias tag. See Create an SSH activity for instructions.
This example uses a VM provisioned on an AWS LDC datacenter.

1. In the Cloud User Portal, click **Launch a Stack** and then select the AWS VM.
2. On the General Info tab, enter a unique and meaningful **Stack Name**, select **AWS Datacenter** for the **Location**, and then click **Next**.
3. On the Provision tab, enter a **Virtual Server Node Name** (the name can be the same as the stack name) and then click **Submit**.
   The Cloud Activities page displays the VM that you requested and the status of your request. The Request Status section displays Deployment Successful when complete.
4. In your instance, view the workflow:
   a) Filter for workflow and then click **Workflow Editor**.
   b) At the right of the workflow list window, filter in the Workflows tab and select the workflow to work with. When the workflow appears in the canvas, you can see the Execute SSH Command activity pill in the workflow.
5. To edit the activities in the workflow, click the workflow properties icon (circle containing an i) and perform the following:
   a) On the **Custom** tab at the right, go to **Custom Activities > Global** and double-click the activity to modify.
      When the new activity opens in the Activity Designer, you can use the form to create a reusable orchestration activity.
   b) To change from the Cloud Management application to the Global application to edit the record, click ‘here’ in the menu bar. The Global application makes the activities available across the entire platform.
6. Edit the fields in each of the following activity tabs. Click **Continue** to move to the next tab.

<table>
<thead>
<tr>
<th>Activity tab</th>
<th>Field descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>Enter a unique name for the activity</td>
</tr>
<tr>
<td>Inputs</td>
<td>Enter values for the host, type, and whether the activity’s mandatory</td>
</tr>
</tbody>
</table>
| Execution Command  | To form the command to be executed, drag inputs from the list to the form. Available inputs:  
   - Host, which is the IP address of the VM to connect to.  
   - Command, which is any script or command single line to be run on the VM.  
   - Credential tag, which is the alias for the sys_id of the VM. You can either enter the credential tag manually or provide an expression in the resource script, which then auto-populates the Credential tag field.  
   Script: $(Script:CMPVMUtil.getCredentialAlias(arg= $(parameter.resourceID))) |
| Output             | Enter the name of each output, and the type of each output, for example, string     |
| Conditions         | Enter any required conditions. Conditions are optional                              |

Use SSH to shutdown a VM using workflow type operation

Use SSH to remotely shutdown a VM using a workflow type operation. Follow this use case to step through all of the tasks required to accomplish this type of shutdown.
Role required: workflow designer or sn_cmp.cloud_admin

1. To create the SSH custom activity, follow these steps:
   a) Navigate to Admin Portal > Workflows > Workflow Editor.
   b) Click the Custom tab, then click + (plus sign) to create a new custom activity.
   c) From the list of types of activities, choose SSH.
   d) The Activity Designer opens with the General tab displayed. Enter a unique name for the activity, then click Continue.
   e) In the Inputs tab, add the input variables with the following steps, then click Continue:
      f) In the Input table, click the + in a circle to create a new input variable.
      g) Click to the right of ABC in the Name column, and type the input name.
         Inputs vary according to the object to be impacted. Here, we’re creating an activity to stop a VM, so the inputs reflect the values needed to accomplish that action.
         The Type and Mandatory columns populate automatically according to input you enter.
      h) Create a new input object for each input in the following table, then click Continue:

<table>
<thead>
<tr>
<th>Input variable</th>
<th>Type</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>String</td>
<td>No</td>
</tr>
<tr>
<td>Command</td>
<td>String</td>
<td>No</td>
</tr>
<tr>
<td>CredentialTag</td>
<td>String</td>
<td>No</td>
</tr>
</tbody>
</table>

   i) On the Execution Command tab, perform the following:
   j) In the Output table, click the + in a circle to create a new output variable.
   k) Click to the right of ABC in the Name column, and type the output name.
      The output variables reflect the possible outcomes when the workflow attempts to SSH to the VM. The Type column populates automatically.

<table>
<thead>
<tr>
<th>Outputs</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>String</td>
</tr>
<tr>
<td>output</td>
<td>String</td>
</tr>
</tbody>
</table>
l) Click and drag the error variable to the Variable name field in the Parsing rules table.
m) In the Parsing rule for error dialog, in the Parsing source drop-down, select the appropriate source; in this case, executionResult.errorMessages.
n) Repeat for the output variable, selecting executionResult.output as the Parsing source, then click Submit.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>String</td>
</tr>
<tr>
<td>output</td>
<td>String</td>
</tr>
</tbody>
</table>

o) Click Continue.
Conditions are optional.
p) Click Save, then click Publish to make the activity available to use in the workflow.

2. To create the SSH workflow needed to power off the VM, follow these steps:
When you create a workflow, you associate it with a table and with an activity.
• You can create the workflow on a table at the global-level or a table on the application-level. Use the global table to make the workflow available to use with all ServiceNow applications. Here, we use a table for the Cloud Management, so this workflow will only be available to use with that application.
• You can create the workflow from the Workflow Editor or from the Activity Designer. Here, the Activity Designer is used.

a) From the Activity Designer, select the Workflows tab at the right, then click the + (plus sign) to add a new workflow.
b) In the New Workflow window, enter a unique name for the workflow.
c) From the Table drop-down, select the table to use. Here, we search for and use the Virtual Machine instance cmdb_ci_vm_instance because we intend to use the workflow to power off a VM.
d) Click Submit.
e) When the workflow displays in the canvas, delete the connection line between the start point and end point of the workflow.
f) Delete the connection line between the start point and end point of the workflow.
g) Right-click the more info canvas and select Edit Inputs.
h) Create the following inputs:

Note: Take the input names from the Column field, not from the Label field. If the workflow is associated with a global-level table, the input name in the Column field...
is prefixed with u_, for example, u_ipaddress. Here, the table is application-level, so it needs no prefix.

<table>
<thead>
<tr>
<th>Type</th>
<th>Label</th>
<th>Column (automatically added)</th>
<th>Max Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>IPAddress</td>
<td>ipaddress</td>
<td>100</td>
</tr>
<tr>
<td>String</td>
<td>CredentialTag</td>
<td>credentialtag</td>
<td>100</td>
</tr>
</tbody>
</table>

i) Click Submit.

j) Right-click the canvas and select Add Custom Activity.

k) In the Versions window, search for and select the activity you created initially in this use case.

l) In the New Activity window for that activity (Workflow Activity New record (Diagrammer view)), enter a name for the activity.

m) In the Host field, type the input mapping that the activity expects. Here, that is the IP address of the host: $(workflow.inputs.u_ipaddress)

   Use this expression to get the IP Address:
   $(Script: CMPVMUtils.getReachableIp(arg=$(Stack.items[Virtual Server].attributes[sys_id])))

n) In the Command field, type shutdown -h now

o) In the Credential tag field, type $(workflow.inputs.u_credentialtag)

   Use this expression to get the credential tag:
   $(Script: CMPVMUtils.getCredentialAlias(arg=${Stack.items[Virtual Server].attributes[sys_id]})

p) Click Submit.

   The activity displays in the workflow canvas.

q) Right-click the canvas, and select Add Core Activity to add a run script activity.

r) In the Workflow Activity Definitions window, search for and select Run Script.

s) In the New Activity: Run Script window, type a name for the activity.

t) Cut and paste a script (appropriate for your environment) in the Script field.

   Example script:
   ```java
   if ( gs.nil(data.get(3).error)) {
       current.state = 'off';
       current.setWorkflow(false);
       current.update();
   }
   ```

u) Click Submit.

v) In the canvas, connect the connectors between the Begin point, the activity, the Run Script, and the End point.

w) Validate the workflow by using the workflow validation.
3. To add the operation to a resource block, following these steps:
   You can select from the existing operations in a resource block to work with your workflow or you can create a custom operation.
   a) In the Cloud Admin Portal, navigate to Design > Resource Blocks.
   b) Select a resource block, for example Virtual Server.
   c) Move the state of the resource block from Published to Draft to make it editable.
   d) Select the Operations tab, then from the Interface drop-down field, select an interface that is available to the user. The default interface is Virtual Server Interface.
   e) Click + to the right of the Operation field, then in the Add Operation dialog, enter a meaningful and unique name.
   f) Select an operation from the Operation Type drop-down.
   g) Select Public from the Access Type field.
   h) Fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Operation Type</th>
<th>Select the operation to perform.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Type</td>
<td>For the user to see the operation at the Resources level during provisioning, select Public. Operations set to Private are not visible during provisioning.</td>
</tr>
</tbody>
</table>

The operation is available on the resource when you select the resource for your workflow.

i) To add a resource operation step to invoke the workflow, still in the Operations tab, click Steps > Add Step.
   To enable the workflow to be exported as part of the update sets, you must create an extension interface and add the new operation. See Extend Cloud Management resource blocks with an override operation for details.

j) In the Add Operation Steps popup, select Invoke Workflow for Operation Type.

k) Select the Workflow from the list of workflows that you created.
   The system can filter the list using tags. To add a tag:
   1. Navigate to System Properties.
   2. Locate and select the system property sn_cmp.workflow_tag_filter.
   3. Edit the property. Add a tag or comma-separated list of tags.
   4. Click Save.

l) To filter the workflows:
   1. On the workflow table, open the workflow.
   2. Click Add Tag.
   3. Enter the value that you provided earlier in the property sn_cmp.workflow_tag_filter.
   4. Click Enter to add the value as a tag to the workflow. The new step appears after the software generates the new Day 2 Operation Catalog.
m) Workflows can exist that are created on tables other than global. You can add a workflow created on such a table. To execute operations on this type of workflow, you need the sys_id of the record in which the workflow is executing to create the current record. So when the designer adds a workflow that is on table other than global, the system creates a `wf_current` parameter in the operation input parameter. You can then write an expression to map the sys_id of the resource in which the operation is executing. Once this is completed, the designer can use the ‘current’ key word in workflow scripts.

To add the workflow:

1. After you create the step, the system adds the workflow inputs to the operation input parameters. You can then provide the mapping for these parameters, if needed.
2. Click the Generate Catalog button to create the catalog item for the operation.
3. After the catalog generates, you can add from the load and field change rules to the catalog items.

n) Set the resource block to the Published state to make the workflow available in the Cloud User Portal.

4. To execute the operation from the User Portal, follow these steps:
   a) Provision a simple AWS VM from the portal.
   b) After the VM provisioning completes, navigate to the stack > VM Resource.
   c) In the Select Resource Operation option, select the custom stop activity you created earlier in this use case, then click OK.
      The operation executes after the RITM is created and the VM state changes in the AWS console.
   d) Navigate to Operate > Trails to view the trail logs and trace the operation steps.

5. To troubleshoot, if needed, follow these steps:
   a) Use the request item (RITM) and navigate to the RootCauseAnalysis Dashboard. The RITM has the link to the workflow context of the currently executed operation.
   b) Check for these common errors:

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Error message</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM shuts down but there has been an issue with the SSH connection</td>
<td>Error: job finished with status ERROR: Problem in SSH session, job aborted: Connection unexpectedly closed by SSH server: \n',</td>
</tr>
<tr>
<td>VM is already shutdown / not able to reach the VM</td>
<td>'Cannot connect, status is TCP_CONNECTION_FAILURE. Timed out while waiting for TCP to connect to 10.198.252.224:22: \n',</td>
</tr>
</tbody>
</table>

**Note:** Confirm that the IP address of the VM in the wf_context is the appropriate address. The VM might have been already shutdown. Confirm the IP address is accessible.
### ServiceNow New York IT Operations Management

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Error message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root access unavailable</td>
<td>'Failed to issue method call: Access denied\nMust be root.\nExit status: 1\n'</td>
</tr>
</tbody>
</table>

**Note:** Confirm that the Must Sudo box is checked in the SSH activity at the beginning of this use case.

Create the SSH custom activity
Create an SSH custom activity so you can use it in your workflow.

Role required: cloud designer or admin

1. Navigate to **Admin Portal > Workflows > Workflow Editor**.
2. Click the **Custom** tab, then click + (plus sign) to create a new custom activity.
3. From the list of types of activities, choose **SSH**.
4. The Activity Designer opens with the General tab displayed. Enter a unique name for the activity, then click **Continue**.
5. In the Inputs tab, add the input variables with the following steps, then click **Continue**:
   a) In the Input table, click the + in a circle to create a new input variable.
   b) Click to the right of ABC in the Name column, and type the input name.
      - Inputs vary according to the object to be impacted. Here, we're creating an activity to stop a VM, so the inputs reflect the values needed to accomplish that action.
      - The Type and Mandatory columns populate automatically according to input you enter.
   c) Create a new input object for each input in the following table, then click **Continue**:

<table>
<thead>
<tr>
<th>Input variable</th>
<th>Type</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>String</td>
<td>No</td>
</tr>
<tr>
<td>Command</td>
<td>String</td>
<td>No</td>
</tr>
</tbody>
</table>
6. On the Execution Command tab, perform the following:
   a) Click and drag the input variables from the left to the template fields at the right. This step 
      creates the expressions that will be used.
   b) Check the Must Sudo box so that you can later log in to the virtual server.
   c) Click Continue.

7. On the Outputs tab, add the output variables with the following steps:
   a) In the Output table, click the + in a circle to create a new output variable.
   b) Click to the right of ABC in the Name column, and type the output name.
      The output variables reflect the possible outcomes when the workflow attempts to SSH to 
      the VM. The Type column populates automatically.

<table>
<thead>
<tr>
<th>Outputs</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>String</td>
</tr>
<tr>
<td>output</td>
<td>String</td>
</tr>
</tbody>
</table>

   c) Click and drag the error variable to the Variable name field in the Parsing rules table.
   d) In the Parsing rule for error dialog, in the Parsing source drop-down, select the appropriate 
      source; in this case, executionResult.errorMessages.
   e) Repeat for the output variable, selecting executionResult.output as the Parsing source, 
      then click Submit.
f) Click **Continue**.

   Conditions are optional.

g) Click **Save**, then click **Publish** to make the activity available to use in the workflow.

You have now created a custom activity that exists as a container until you add in the input mappings to it. The mappings tell the activity where the information it needs is coming from. You enter the input mappings during the workflow creation procedure. See [*Create an SSH workflow*](#).

### Create an SSH workflow

Create the SSH workflow needed to power off the VM. Use the SSH custom activity that you already created.

- **Role required:** workflow designer or admin
- **Have an SSH custom activity already created**

When you create a workflow, you associate it with a table and with an activity.

- You can create the workflow on a table at the global-level or a table on the application-level.
  - Use the global table to make the workflow available to use with all ServiceNow applications.
  - Here, we use a table for the Cloud Management, so this workflow will only be available to use with that application.
- You can create the workflow from the Workflow Editor or from the Activity Designer. Here, the Activity Designer is used.

1. From the Activity Designer, select the **Workflows** tab at the right, then click the + (plus sign) to add a new workflow.

2. In the New Workflow window, perform these steps:
   a) Enter a unique name for the workflow.
   b) From the **Table** drop-down, select the table to use. Here, we search for and use the **Virtual Machine instance cmdb_ci_vm_instance** because we intend to use the workflow to power off a VM.
   c) Click **Submit**.

3. When the workflow displays in the canvas, perform these steps:
   a) Delete the connection line between the start point and end point of the workflow.
   b) Right-click the more info canvas and select **Edit Inputs**.
   c) Create the following inputs:
Note: Take the input names from the Column field, not from the Label field. If the workflow is associated with a global-level table, the input name in the Column field is prefixed with **u_**, for example, **u_ipaddress**. Here, the table is application-level, so it needs no prefix.

<table>
<thead>
<tr>
<th>Type</th>
<th>Label</th>
<th>Column (automatically added)</th>
<th>Max Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>IPAddress</td>
<td>ipaddress</td>
<td>100</td>
</tr>
<tr>
<td>String</td>
<td>CredentialTag</td>
<td>credentialtag</td>
<td>100</td>
</tr>
</tbody>
</table>

d) Click Submit.

4. Right-click the canvas and select **Add Custom Activity**.

5. In the Versions window, search for and select the activity you created earlier in **Create the SSH custom activity**.

6. In the New Activity window for that activity (Workflow Activity New record (Diagrammer view)), perform these steps:
   a) Enter a name for the activity.
   b) In the Host field, type the input mapping that the activity expects. Here, that is the IP address of the host: **${workflow.inputs.u_ipaddress}**
      Use this expression to get the IP Address:
      ```javascript
      $(Script:CMPVMUtils.getReachableIp[arg=$(Stack.items[Virtual Server].attributes[sys_id])])
      ```
   c) In the Command field, type **shutdown -h now**
   d) In the Credential tag field, type **${workflow.inputs.u_credentialtag}**
      Use this expression to get the credential tag:
      ```javascript
      $(Script:CMPVMUtils.getCredentialAlias[arg=${Stack.items[Virtual Server].attributes[sys_id]}])
      ```
   e) Click Submit.
      The activity displays in the workflow canvas.

7. Right-click the canvas, and select **Add Core Activity** to add a run script activity.

8. In the Workflow Activity Definitions window, search for and select **Run Script**.

9. In the New Activity: Run Script window, perform these steps:
   a) Type a name for the activity.
   b) Cut and paste a script (appropriate for your environment) in the Script field.
      This script updates the state of the VM after it is shut down.
      Example script:
      ```javascript
      if ( gs.nil(data.get(3).error) ) {
        current.state = 'off';
        current.setWorkflow(false);
        current.update();
      }
      ```
© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
c) Click **Submit**.

10. In the canvas, connect the connectors between the Begin point, the activity, the Run Script, and the End point.
11. Click **Publish** to make the workflow available.
12. Validate the workflow by using the workflow validation.

**Note:** The system does not recommend the use of current.update(). This can be ignored.

Map credentials between Cloud Management and Terraform
You can map credential and service account fields between Cloud Management and Terraform using the IaC Provider Settings. This enables you to store and manage credentials in an external credential repository rather than directly in a ServiceNow credentials record.

Role required: sn_cmp.cloud_admin; Cloud Management: Terraform Cloud Connector (v1.06) must be installed.

You can use the (Infrastructure as Code) IaC Provider Settings user interface to perform credential and service account mappings for DataCenter Credentials used by the Terraform provider to provision datacenter resources and Cloud Management for discovery.

1. Navigate to **IaC Provider Settings > Terraform**, and click **New**. The **Terraform Datacenter Mapping** form appears.
2. Complete the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Terraform Provider</td>
<td>Name of the Terraform Provider.</td>
</tr>
<tr>
<td>Datacenter Block</td>
<td>Select the Resource Block whose credentials you are mapping from the lookup table.</td>
</tr>
<tr>
<td>Credential Type</td>
<td>Select the appropriate credential type from the lookup table. For example, Azure Service Principal (azure_service_principal).</td>
</tr>
<tr>
<td>Credential Resolver</td>
<td>Select an appropriate MID Server script from the lookup table. For example, Search and select the TerraformProviderCredentialResolver script.</td>
</tr>
</tbody>
</table>

3. Click **Submit**. The Terraform provider credentials record is created and displayed in the Terraform Datacenter Mappings list.

4. On the record you just created, select the **Credentials Mappings** tab, and then click **New**.  
   a) Complete the **Credential Mapping** form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider Authentication Key</td>
<td>Authentication key (variable) field name as used by the provider. For example, Terraform provider block for AWS uses the field secret_key.</td>
</tr>
</tbody>
</table>
5. Select the Service Account Mappings tab, and click New.
   a) Complete the Service Account Mapping form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Credential Key</td>
<td>The corresponding field in the Discovery credentials from the lookup table. For example, access_key field in Terraform corresponds to Secret key field in ServiceNow.</td>
</tr>
<tr>
<td>Terraform Datacenter Mapping</td>
<td>Datacenter for which you are mapping the credentials.</td>
</tr>
</tbody>
</table>

   b) Click Submit.

The field values you added under the Credential Mapping and Service Account Mapping tabs are mapped.

Create a cloud catalog item
Create a cloud catalog item for provisioning, based on a template or a blueprint, and publish the catalog item to provide a service.

Role required: sn_cmp.cloud_service_designer.

While creating a catalog item, you can associate the catalog item with either a template or an existing blueprint. After you generate a catalog item, the system automatically generates components such as variable sets, lifecycle operations, rules, and pre- and post-provision operations. You can manage all these components from the Create Catalog Item page.

Note: You can create blueprints first and then generate a catalog item from the blueprint.

1. In the Cloud Admin Portal, navigate to Design > Cloud Catalog items.
   All the catalog items that you’ve created are listed in a card layout view. You can click an inactive catalog item to provision it.
2. Click New.
   The Cloud Catalog Items screen appears.
3. On the form, fill in the fields.
Cloud Catalog Item form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the catalog item.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for making the catalog item available. This check box can be selected only after you've selected a blueprint.</td>
</tr>
<tr>
<td>Icon</td>
<td>Image to appear as an icon beside the item name in the catalog. You can upload a 16x16 pixel image. If no image is uploaded, the default icon appears beside this item. The uploaded image overwrites the default image. The image you upload appears in the service catalog, not in the ServiceNow Cloud Management application.</td>
</tr>
<tr>
<td>Picture</td>
<td>Image for the catalog item. The image that you select is displayed for the catalog item in the Cloud User Portal (after you click Launch a Stack).</td>
</tr>
<tr>
<td>Catalogs</td>
<td>Names of catalogs.</td>
</tr>
<tr>
<td>Category</td>
<td>Category for the item appears by default. You can change the category. Catalog searches find only items that are assigned to a category. The categories appear in the service catalog. The catalog item can appear in multiple categories. To add categories, click the Categories tab and then click New or Edit.</td>
</tr>
<tr>
<td>Source</td>
<td>Source type (Cloud Template or Custom) to associate the catalog item with.</td>
</tr>
<tr>
<td>Template type</td>
<td>If you selected Cloud Template as your source, then select the type of template (ARM Template or CloudFormation Template).</td>
</tr>
<tr>
<td>Blueprint</td>
<td>If you selected Custom in the Source field, then select a blueprint to associate with the catalog item.</td>
</tr>
<tr>
<td>Blueprint Operation</td>
<td>This field appears only when you select a blueprint. After you generate the catalog item, the value in this field is automatically populated.</td>
</tr>
<tr>
<td>Short description</td>
<td>Brief description of the catalog item. This description appears in the Cloud User Portal next to the catalog item.</td>
</tr>
<tr>
<td>Description</td>
<td>Full description of the catalog item. This description appears in the service catalog; not in the Cloud Management application. You can embed videos, images, links to internal knowledge base (KB) articles, and links to external sources of information and instruction documentation.</td>
</tr>
</tbody>
</table>

4. Click Submit.
The catalog item and all associated components such as variable sets, lifecycle operations, and rules are generated and appear as tabs in the Cloud Catalog Item page.

**Note:** If you selected Template from the Source type list, you must create a template and associate the template with the catalog item.

The related links are actions that you can perform on the catalog item.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Export Catalog Item</td>
<td>Exports all the components (such as blueprints, resource blocks, rules) of the catalog item from one environment to the other. See <a href="#">Moving Cloud Management content across environments</a>.</td>
</tr>
<tr>
<td>Item Diagnostic</td>
<td>Identifies possible issues with the catalog item data and configuration. See <a href="#">Item diagnostic report</a>.</td>
</tr>
<tr>
<td>Manage Attributes</td>
<td>See <a href="#">Manage operation attributes</a>.</td>
</tr>
<tr>
<td>Reset Catalog Item</td>
<td>Resets (to the last change you made) any customizations that you've made to the catalog item.</td>
</tr>
</tbody>
</table>

**Create a cloud template.**

Publish a blueprint as a cloud catalog item

You can create and publish a catalog item directly from a blueprint.

Role required: sn_cmp.cloud_service_designer

1. In the Cloud Admin Portal, navigate to Design > Blueprints.
2. Open a blueprint (in draft mode), click the Catalog tab, and then click Create Catalog Item. The Create Blueprint Catalog dialog box appears.
3. Enter a name for the catalog item in the Name field and click Save. The catalog item is generated and appears in the Catalog tab.
4. Click the catalog item to make any updates to the catalog item. The Cloud Catalog Item screen appears.
5. Click Update to save the changes you made.

**Create a cloud template**

Create a cloud template and associate the template with a catalog item. Once you’ve created a template, you can reuse the template to quickly create additional catalog items for the services you want to provision.

Role required: sn_cmp.cloud_service_designer.

You can create a template by uploading a generic template file, importing a template from a public URL, or by copying and pasting the contents of an existing template. AWS CloudFormation templates support JSON and YAML. ARM supports JSON.

**Note:** Catalog items that were based on templates in earlier releases are now treated as if they are based on blueprints.

You can update a template as often as needed. With each update a new version of the template is created. A template can be in one of three states: draft, active, or history. When you
create a template and do not activate it, the template is in a draft state. Once you activate the template, the state changes to active. If you create another template and activate it, the state of the previous active template turns into history. At any given time, only one template is in active state. Each time you activate a template, the previous active template becomes history.

1. In the Cloud Admin Portal, navigate to **Design** > **Cloud Catalog Items** and open the catalog item.

2. Click **Cloud Templates** > **New** and then enter a meaningful short description in the **Short description** field.

3. Specify the **Ingestion method**.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import from URL</td>
<td>Import a template by specifying a public URL where the template resides. Select this option and then click the lock icon (🔒) to unlock the Cloud template URL field. Enter the public URL in this field.</td>
</tr>
</tbody>
</table>

**Note:** For the public URL, only HTTP and HTTPS protocols are supported. FTP is not supported. Ensure that the size of the template should not exceed the default value of 3 MB. You can change the default value of the file size by changing the value in the `sn_cmp.template_content_size_supported_inbytes` property in the `sys_properties` table. Ensure that you enter the file size in bytes. There’s also a default time-out of 5 minutes for an HTTP request. You can change the time-out value in the `sn_cmp.template_url_import_http_timeout` property in the `sys_properties` table. Ensure that you enter the new value in milliseconds.

<table>
<thead>
<tr>
<th>Upload a file</th>
<th>Upload a template from your local workstation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use template body</td>
<td>Paste the contents of the template file in the <strong>Body</strong> field.</td>
</tr>
</tbody>
</table>

After you specify the **Ingestion method**, the template is validated and the results appear in the **Validation status** and **Validation message** fields.

4. Click **Submit**.

The template is created and the Cloud Catalog Item page opens.

5. If this is the first version of template, skip this step. On the **Cloud Templates** tab, open the template (currently in Draft state) to view the template version parameters.

**Note:** The first version of a template that you create and publish has no conflict issues. If you update the template, a conflict can arise. If a parameter **Action Type** has the value **Update**, then specify a value for **Decision**.
6. Open each parameter with a **Decision** value of **Pending**, select a value, and then click the check mark.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skip Update</td>
<td>Discards the updates to the template.</td>
</tr>
<tr>
<td>Use template</td>
<td>Applies the updates to the template.</td>
</tr>
</tbody>
</table>

7. After you update each **Pending** decision, click **Activate**. The state of the template changes from **Draft** to **Active** and the system generates the catalog item.

Create pre-provisioning operations

Pre-provisioning operations are invoked on catalog items after you submit a request and before the catalog item is provisioned. You can specify workflows, scripts, CAPI calls, or resource operations to run automatically before the catalog item is provisioned. For example, you might want to send out notifications that provisioning of the item is about to begin. Setting up an operation as a pre-provision operation means you don’t have to do any additional steps to invoke the operation, saving you time and effort.

Role required: sn_cmp.cloud_service_designer.
You can create pre-provisioning tasks only for catalog items that are based on templates.

1. In the Cloud Admin Portal, navigate to **Design > Cloud Catalog Items**.
2. Open a catalog item, click the **Pre Provisioning Operation** tab, click **New**, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step Type</td>
<td>Pre-provisioning operation. Based on the type of pre-provisioning operation you select, additional fields appear.</td>
</tr>
<tr>
<td></td>
<td>· CAPI:</td>
</tr>
<tr>
<td></td>
<td>· CAPI Provider</td>
</tr>
<tr>
<td></td>
<td>· CAPI Method</td>
</tr>
<tr>
<td></td>
<td>· CAPI Version</td>
</tr>
<tr>
<td></td>
<td>· InstanceScript:</td>
</tr>
<tr>
<td></td>
<td>· Script: Enter the name of the script.</td>
</tr>
<tr>
<td></td>
<td>· Resource:</td>
</tr>
<tr>
<td></td>
<td>· Resource Interface</td>
</tr>
<tr>
<td></td>
<td>· Interface Operation</td>
</tr>
<tr>
<td></td>
<td>· Workflow:</td>
</tr>
<tr>
<td></td>
<td>· Flow: Select a workflow.</td>
</tr>
<tr>
<td>Order</td>
<td>Order of pre-provisioning operation. Enter the order that pre-provisioning operation should be invoked. Lower numbers are invoked first. For example, if you enter 2 for CAPI and 3 for InstanceScript, then CAPI is invoked first and then the InstanceScript gets invoked. If you do not enter any value in this field, the system automatically enters a value in increments of 20.</td>
</tr>
<tr>
<td>Enabled</td>
<td>Option for enabling the pre-provision operation you want invoked.</td>
</tr>
</tbody>
</table>

3. Click **Submit**.
Parameters for the pre-provisioning operations are created as operation attributes in the Manage Attributes page.

4. To add values for the pre-provision operation attributes, click **Manage Attributes**. The Manage Attributes screen appears.

5. Click **Pre Provision**.
All attributes for the pre-provision operation are listed.
6. You can either create expression mappings for an attribute (in the Mapping field) or generate variables for an attribute. See Manage operation attributes for more information.

7. Click Apply Changes.
   All the configurations you made to the attributes are saved and you are returned to the Cloud Catalog Items screen.

Create or update a catalog item from a Google Cloud Deployment Manager (GDM) template
Instead of generating a catalog item based on a blueprint, you can generate a catalog item based on a template that is held in the GDM configuration management system. The system parses the attributes in both the template file and optional supporting files and then generates a blueprint and a catalog item.

Role required: sn_cmp.cloud_service_designer

Cloud Management supports Google Cloud Platform templates managed by Cloud Deployment Manager (GDM) that were created in YAML, Python, or Jinja2.

The procedure to create a catalog item from a template involves the following steps:

1. Create and save a catalog item record that references the template. The catalog item is in Draft state. In a later step, you activate the catalog item to make it available to cloud users.

2. Create a cloud template record. The cloud template specifies a configuration installable (a directory structure that includes the template file) to use as the source of the catalog item.

   Note: You can update the cloud template record later, by referencing an updated configuration installable. Instructions are included in this procedure.

3. Google Cloud Platform only: If the template file depends on supporting files (typically updates or template code that specifies attribute settings), you add the files to the Dependent Templates related list.

4. Activate the cloud template to generate a resource block and blueprint that are associated with the catalog item and to make the cloud template available for use by the catalog item.

5. Activate the catalog item so that cloud users in the Cloud User Portal can use it to provision resources.

1. In the Cloud Admin Portal, navigate to Design > Cloud Catalog Items.
2. Click New and then enter a unique and meaningful Name, Short description, and Description.
3. Specify the Source as Cloud Template and the Template Type as Google Cloud Deployment Manager.
4. Right-click in the header and select **Save**.
5. On the Cloud Templates related list, click **New** and enter a unique **Name** and **Short description**.
6. Specify the content of the template. You have three options:

- Paste the text of the template file into the form:
  1. Set **Ingestion method** to **Use Template Body**.
  2. You typically use either a custom template that you created or a template in a public GitHub or Bitbucket repository. Open the template file in a text editor and copy the full text of the file.
  3. On the Cloud Template Versions form, paste the text into the **Body** text box.
- Specify the URL of the template file:
1. Set **Ingestion method** to **Import from URL**.
2. In a browser, go to the template file. Copy/paste the full URL of the template into the **Cloud template URL** text box.

- **Upload the template file:**
  1. Set **Ingestion method** to **Upload a file**.
  2. Click the **Click to add** link for the **Upload file** field.
3. Click **Choose File**, select the file, and then click **OK**. The system adds the template and the filename appears in the **Upload file** field. The results of the import process appear in the **Validation status** and **Validation message** fields.

7. The template code appears in the **Body** text box. If the file does not include a path entry under `imports:`, then skip this step. If the file does include an entry, then the template file depends on at least one dependent file (typically updates or files that specify attribute settings). Perform the following procedure for each dependent file (`cloudsql.jinja` in the example):

   a) In a text editor, open the dependant template file (`cloudsql.jinja` in the example).
   b) Copy the full text of the file.
   c) On the **GDM Dependent Templates** related list, click **New**.
d) Paste the full text of the dependent template file into the **File content** text box.

e) Enter the exact name of the dependent template file into the **File name** text box.

![GDM Dependent Templates form]

f) Click **Submit**. The file appears in the **GDM Dependent Templates** related list.

g) Repeat the procedure for each dependent template file.

8. On the Cloud Template Versions form, click **Activate**.
   The system generates the catalog item and a blueprint with the name that you specified for
   the catalog item.

9. On the Cloud Catalog Item form, select the **Active** check box, and then click **Submit**.
   The system displays the catalog item on the Cloud Catalog Items page and publishes the
   catalog item to the Cloud Service Catalog for use by end users.
Verify that the catalog item appears in the Cloud User Portal.

Create or update a catalog item based on a Terraform template

Instead of generating a catalog item based on a blueprint, you can generate a catalog item based on a template that is held in the Terraform configuration management system. The system parses the attributes in the template files and then generates a record of the template data, blueprint, and resource block. You can then use the template record as the basis for a catalog item.

Role required: sn_cmp.cloud_service_designer

IBM Cloud Connector, VMware vSphere, and Microsoft Azure Cloud support Terraform template files in .tf and .tf.json formats.

The procedure to create a catalog item from a template involves the following steps:

1. Create and save a catalog item record that references the template. The catalog item is in Draft state. In a later step, you activate the catalog item to make it available to cloud users.

2. Create a cloud template record. The cloud template specifies a configuration installable (a directory structure that includes the template file) to use as the source of the catalog item.

   Note: You can update the cloud template record later, by referencing an updated configuration installable. Instructions are included in this procedure.

3. Google Cloud Platform only: If the template file depends on supporting files (typically updates or template code that specifies attribute settings), you add the files to the Dependent Templates related list.

4. Activate the cloud template to generate a resource block and blueprint that are associated with the catalog item and to make the cloud template available for use by the catalog item.
5. Activate the catalog item so that cloud users in the Cloud User Portal can use it to provision resources.

1. Create and save a catalog item record.
   a) In the Cloud Admin Portal, navigate to **Design > Cloud Catalog Items**.
   b) Click **New**, enter a unique and descriptive **Name**, fill in the form, and then right-click in the header and select **Save**.

<table>
<thead>
<tr>
<th>Source</th>
<th>Select Configuration Management Template.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider type</td>
<td>• Select Terraform Environment if Terraform is running on a Linux host.</td>
</tr>
<tr>
<td></td>
<td>• Select Terraform Windows Environment if Terraform is running on a Windows host.</td>
</tr>
<tr>
<td>Config provider</td>
<td>Select the appropriate config provider.</td>
</tr>
<tr>
<td></td>
<td>For instructions on creating a Terraform config provider record, see <strong>Create a Terraform config provider and run Discovery.</strong></td>
</tr>
</tbody>
</table>

   The catalog item record is created.

2. Create a cloud template that references a Terraform configuration installable.
   a) On the **Cloud Templates** related list, click **New**.
   b) Select the appropriate Terraform **Configuration installable** and then right-click in the header and select **Save**.

   The system creates the cloud template record, reports success, and displays the content of the configuration installable. Notice that the cloud template is in the **Draft** state. In the next step, you activate the cloud template to make it available for use by the catalog item.

   **Note:** A catalog item can include multiple versions of the cloud template but only one version can be active.
Note: Skip the next two substeps if you are defining a new cloud template record. Configuration conflicts can arise when you create a cloud template with an updated version of a Terraform configuration installable.
c) On the **Cloud Templates** tab, click the template to view the Template Version Parameters related list. If a parameter **Action Type** has the value **Update**, then you must specify a **Decision** to use the current parameter value in the cloud template or the value in the updated configuration installable.

<table>
<thead>
<tr>
<th>Template Version Parameters</th>
<th>Template Version Parameters</th>
<th>GDM Dependent Templates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Template version = 3</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td><strong>Decision</strong></td>
<td><strong>Action type</strong></td>
</tr>
<tr>
<td><strong>adminUsername</strong></td>
<td><strong>Type</strong></td>
<td><strong>DataSource</strong></td>
</tr>
<tr>
<td><strong>Use Template</strong></td>
<td><strong>Use Template</strong></td>
<td><strong>DataSource Value</strong></td>
</tr>
<tr>
<td><strong>New</strong></td>
<td><strong>New</strong></td>
<td><strong>Default Value</strong></td>
</tr>
</tbody>
</table>

| **windowsOSVersion**        | **Use Template**            | **Update**              |
| **DataSource**              | **DataSource Value**        | **CSVList**             |
| **String**                  | **String**                  | **Datacenter,2012-R2,** |
| **2006-R2-SP1,**            | **Text**                    | **2012-R2,**            |
| **Datacenter,2012-R2,**     |                             | **Datacenter,2016,**    |
| **2016-Datacenter**         |                             |                         |

d) On the **Template Parameters** tab, specify the parameters that end users must specify when requesting a stack by selecting **Yes** in the **Create Param** field. Parameters with a value of **No** are not required when requesting a stack.

3. Click **Activate** to activate the cloud template.
   The system generates the resource block and blueprint/template that are associated with the catalog item and then makes the cloud template available for use by the catalog item. The Cloud Catalog Item form opens and displays a message that confirms that the version of the cloud template record is active.

4. On the Cloud Catalog Item form, select the **Active** check box and click **Submit**.
   The catalog item appears in the Cloud User Portal and cloud users can request the catalog item to provision resources.

**Note:** As with any catalog item, you can add or remove a user-viewable field. Click the card to open the catalog item form. On the **Variable Sets** tab, edit the variables.

Verify that the catalog item appears in the Cloud User Portal.

**Create a Terraform config provider and run Discovery**

A config provider is a secure record on your instance that stores the credential and access information for a particular configuration management server (for example, a host running Ansible, or Terraform). Discovery uses the information in the config provider record to access the server to get information on each resource template in the repository. You use the templates to generate catalog items that cloud users can order from the Cloud User Portal.

- Cloud Management role required: cloud_admin
• A host running the open-source version of the Terraform configuration management software must exist.
• On the host that runs Terraform, copy the following values for use in this procedure: Username, Password, SSH Passphrase, SSH private key.

The procedure involves the following steps:
1. Create a Credentials record to securely store Terraform credential information. You refer to the Credentials record when you create a config provider record.
2. Create a config provider.
3. Identify the resources on the Terraform configuration management server.

1. Create a Credentials record:
   a) On your instance, navigate to Orchestration > Credentials and then click New.
   b) Click SSH Private Key Credentials and then fill in the form.

<table>
<thead>
<tr>
<th>Name</th>
<th>Descriptive name for the set of credentials — often the name of the host that is running Terraform. Ensure that the name is unique because you might eventually use multiple Terraform repositories.</th>
</tr>
</thead>
<tbody>
<tr>
<td>User ID</td>
<td>User name for the host that is running Terraform.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for the host that is running Terraform.</td>
</tr>
<tr>
<td>SSH Passphrase</td>
<td>Passphrase for the access credentials for the host that is running Terraform.</td>
</tr>
<tr>
<td>SSH private key</td>
<td></td>
</tr>
</tbody>
</table>
SSH private key | Key for the access credentials for the host that is running Terraform.
---|---
Active | Select the check box to make the credentials available for use.
Applies to | MID Server that uses these credentials for communications.
Order | Order of execution. Discovery tries credentials in sequence — credentials with lower Order values before credentials with higher Order values. Ordering credentials improves performance when there are many credentials, with credentials some used more frequently than others. For example, five credentials out of a total of 20 are used 90% of the time. Configure the five credentials with lower order numbers to try them first.

2. In the Cloud Admin Portal, navigate to Manage > Config Management.
3. Click New, fill in the form, and then click Submit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for the Terraform host, for example, Terraform Config Provider-1.</td>
</tr>
</tbody>
</table>
| Provider | • Select Terraform Environment if Terraform is running on a Linux host.  
• Select Terraform Windows Environment if Terraform is running on a Windows host. |
| URL | Full URL or IP address of the Terraform host. |
| Server Type | Select Linux or Windows. |
| Credential | Login credential for the Terraform host. |

The config provider record is created and appears as a card on the Config Providers page.
4. Click the provider card. To identify the resources on the Terraform configuration management server, click **Discover Now**.

**Note:** You cannot schedule Discovery for configuration management providers. To get updated resource data from a provider, you must run **Discover Now** from this form.

The Terraform config installables contain the workspace and configuration templates.
The terr details appear after provisioning.
Table view of a configuration template:
Config Management

Table view of a workspace:

```json
provider "ibm" {
  region = "us-south"
  bluemix_api_key = "ODr3v7f5_
  softlayer_username = "SL_
  softlayer_api_key = "80a2_
}
resource "ibm_security_group" "sg1" {
  name = "sg1"
  description = "allow my app traffic"
}
```
You can create catalog items from the config installable.

Create post-provision operations

Create post-provision operations that are invoked on newly provisioned catalog items. You can specify workflows, scripts, CAPI calls, or resource operations. For example, you might want to specify a script that runs on the catalog item after the resource is provisioned. Setting up an operation as a post-provision operation means you don’t have to do any additional steps to invoke the operation, saving you time and effort.

Role required: sn_cmp.cloud_service_designer.

You can create post-provision operations only for catalog items that are based on templates.

1. In the Cloud Admin Portal, navigate to Design > Cloud Catalog Items.
2. Open the catalog item, click the Post Provisioning Operation tab, and then click New. The Operation Step form appears.
3. On the form, fill in the fields.
### Operation Setup form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Step Type   | Post-provision operation. Based on the post-operation you select, additional fields appear.  
  - CAPI:  
    - CAPI Provider  
    - CAPI Interface  
    - CAPI Method  
    - CAPI Version  
  - InstanceScript:  
    - Script: Enter the name of the script.  
  - Resource:  
    - Resource Interface  
    - Interface Operation  
  - Workflow:  
    - Flow: Select a workflow. All workflows created in the instance are listed. |
| Order       | Order of post-provision operations. Enter the order (in numerals) in which you want the post-provision operation to be invoked. For example, if you enter the value 2 for CAPI and the value 3 for InstanceScript, then CAPI is invoked first and then the InstanceScript gets invoked. If you do not enter any value in this field, the system automatically enters a value in increments of 20. |
| Enabled     | Option for enabling the post-provision operation you want invoked. |

4. Click **Submit**.  
Parameters for the post-provision operation are automatically created as operation attributes on the Manage Attributes form.

5. To set attribute values for a post-provision operation:
   a) Click the **Manage Attributes** related link.  
      The Manage Attributes form opens.
   b) Select the post-provision operation to update.
      All attributes for the selected operation appear in the Operations list. You can either create expression mappings (shown in the **Mapping** column) for an attribute or can you generate variables for an attribute. See **Manage operation attributes** for more information.
c) On the Operation Attributes form, edit the values in the **Mapping** field as needed.

d) To set a value and not show the field to the cloud user, set **Generate Variable** to false. For VMware post-provision operations, you can set the following attribute values:
Post-provision operation attributes for VMware

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>&lt;Value field&gt; to set in the Mapping expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>NodeCredential</td>
<td>$(capiResolver.NodeCredentialResolver#nodeCredentialId=sysID)</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> You must create the credentials manually to be able to specify the sys_id in the expression</td>
</tr>
<tr>
<td>NodeAddress</td>
<td>$(Script:CMPVMUtils.getReachableIp[arg=$(Stack.items&lt;VM Name&gt;.attributes[sys_id])])</td>
</tr>
<tr>
<td>ScriptExpression</td>
<td>${CloudScript.scripts.&lt;Script Name&gt;}</td>
</tr>
</tbody>
</table>

e) Click **Apply Changes**.
All configurations are saved and the Cloud Catalog Items page opens.

Add post-provisioning to a Cloud Formation Template-based catalog item
Create a workflow post-provisioning operation on a Cloud Formation Template (CFT)-based catalog item.

Role required: sn_cmp.cloud_service_designer.

In this example, create a key pair at the AWS console, then select the appropriate credentials, discover cloud accounts, create a CFT-based catalog item, add a workflow post provisioning operation to the catalog item, and lastly provision the catalog item.

Ensure that you’ve met the following prerequisites before you start working on this example:
- The Cloud Management plugin is activated.
- AWS Credentials and Service Account are set up.
- The Cloud Account for AWS Logical Data Center is created.
- The Key pair is configured on AWS and is available in CMP as “cmdb_ci_cloud_key_pair”.
- A workflow is available on an instance. This example uses the SSH_Workflow_New workflow.
- A script include to fetch the private and public IP is available.

1. At the AWS console, create a new key pair. Give the key pair a name, for example, ItappWOP. When you create a key pair, a PEM file is automatically downloaded onto your system.
2. Open the PEM file and copy all the contents of the file.
3. Navigate to the Credentials application using the filter navigator in your instance.
4. Click **New** and then select **SSH Private Key Credentials**.
5. In the **Name** field, enter a name (Itappwop).
6. In the **SSH Private Key** field, paste the contents of the PEM file.
7. In the **Credential alias** field, enter an alias name (Itappwop).
   The Alias name must be the same name that you entered in the **Name** field.
8. Click **Submit**.
9. To discover cloud accounts to sync data from AWS, run ServiceNow® Discovery.
10. Create a cloud catalog item based on AWS CFT to provision a stack.
   a) Navigate to **Cloud Admin Portal > Design > Cloud Catalog items**
b) Click **New**.
   The Cloud Catalog Item screen appears.

11. Enter a unique name for the catalog item in the **Name** field, select **Cloud Template** in the **Source** field and **Cloud Formation Template** in **Template type** field.

12. Click **Submit**.
   The catalog item is generated.

13. Create a cloud template and associate the template with the catalog item.
   a) Navigate to **Cloud Admin Portal > Design > Cloud Catalog Items**.
   b) Open the catalog item record that you want to create a cloud template for and click **Cloud Templates > New**.
      The Cloud Template Versions screen appears.
   c) In the Ingestion Method list, select **Use template body** and paste the contents of the AWS CFT in the **Body** field.
   d) Click **Save**.
      All variables extracted from the AWS CFT appear in the Template Version Parameters section.
e) Click **Activate**.
The catalog item is generated.
14. Go to the Cloud User Portal and open the catalog item that you just generated. On the Provision tab, you can see all variables extracted from the template. You can add a workflow post provision operation to the catalog item.


16. Open the catalog item record that you want to create the workflow post provisioning operation for.

17. Click the Post Provisioning Operation subtab and then click New. The Operation Step screen appears.

18. Select Workflow in the Setup Type list and select SSH_Workflow_New in the Flow list.

19. Click Submit. Parameters for the workflow operation are automatically created as operation attributes in the Manage Attributes screen.

20. Click Manage Attributes. The Manage Attributes screen appears.


22. Click the key_name attribute and enter either the Credential alias value (Itappwop), or enter an expression mapping in the Mapping field. Mappings specify where to pull the information in the system. See Using expressions in Cloud Management.

23. Click the Stack ID attribute and enter the expression mapping $(context.order.stack).

24. Click Apply Changes. Workflow is mapped as a post provision operation to the catalog item that you generated.

25. Provision the CFT-based catalog item in the Cloud User Portal.
   a) Navigate to Cloud User Portal and click Launch a Stack.
   b) Open the catalog item that you want to provision and enter values in all the mandatory fields on the General Info and Provision tabs.

26. To provision the catalog item, click Submit.

27. Validate the provisioned catalog item upon completion.

Create a variable set for Cloud Management

Variable sets allow you to create a collection of structured variables that can be reused across multiple catalog items and order guides.

Using variable sets saves time because you do not have to create the same variables individually for many catalog items. Also, when variables need to be modified, you can modify the variable set and the changes are reflected across all the catalog items that are associated with the variable set. Variable sets also allow you to define catalog client scripts and UI policies that are applicable to the variables in the set.

When you generate a catalog item, two variable sets, General Info and Provision, are automatically created. You can create as many variable sets as needed.

- Role required: sn_cmp.cloud_service_designer

1. Open a cloud catalog item and in the Variable Sets tab, click New. The Variable Set screen appears.

2. Fill in the form details (as shown in the table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Enter a name for the variable set</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Internal Name | Enter a variable set name for internal use. 
**Note:** If this field is empty, its value is auto-populated based on the **Title** field for all variable types except **break**, **container split**, and **container end**.

Layout | Select a layout display: **1 Column Wide**, 2 **Columns Wide alternating sides**, or **2 Columns Wide, one side, then the other**.

Order | Enter an order number for the variable set.

Description | Enter a description for the variable set.

---

3. **Click Submit.**  
   The new variable set is created and appears in the **Variable Sets** tab. Open the new variable set and click **New** in the **Cloud Variables** tab to create variables for the set.

---

### Edit a variable for Cloud Management

Edit an existing cloud variable or create a new cloud variable to gather information for ordering a catalog item.

Role required: **sn_cmp.cloud_service_designer**

You can edit a variable to change its type, for example from **Masked** to **Lookup Select Box**. You can also change the variable set of a variable, its order, make it active in active, change its permission and availability options, and many more changes.

1. In the Cloud Admin Portal, navigate to **Design > Cloud Catalog Items**.
2. Open a catalog item, click the **Variable Sets** sub tab and click the variable set to which the variable to edit belongs to.  
   All the cloud variables for the selected variable set appear under the **Cloud Variables** sub tab.
3. Click the variable set that you want to edit.  
   The Cloud Variable form appears.
4. On the form, fill in the fields.

#### Cloud Variable form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Type | Variable type to create.  
If you select **Lookup Select Box**, the **Type Specifications** subtab displays these fields:
  * **Use Pool Filter:** Select this check box to add a pool filter to the field.  
  * **Pool:** Select a pool to offer the cloud service user a list of values from an existing table in the catalog order form. For example, you can select **CloudAccountPool** to get a list of all the cloud accounts.  
  * **Pool Filter:** Select a filter to further narrow down the choices in the pool.  

---

© 2020 ServiceNow, Inc. All rights reserved.  
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Field
### Description

<table>
<thead>
<tr>
<th>Variable Set</th>
<th>The variable set that the variable belongs to.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mandatory</td>
<td>Option to make the variable mandatory as part of the ordering process.</td>
</tr>
<tr>
<td>Active</td>
<td>Option to make the variable available for use.</td>
</tr>
<tr>
<td>Order</td>
<td>Order that the variable is placed on the page for the catalog item. The variables are organized from top to bottom, from the least to greatest order value. For example, a variable with an order value of 1 is placed above other variables with higher-order values.</td>
</tr>
</tbody>
</table>

### Questions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question</td>
<td>Question to ask users ordering the catalog item. <strong>Note:</strong> If you select Select Box in the Type field, a Question Choices subtab appears at the end of the screen. You can enter a specific list of choices for the question.</td>
</tr>
<tr>
<td>Name</td>
<td>A name to identify the question. <strong>Note:</strong> If this field is empty, its value is auto-populated based on the Question field for all variable types except break, container split, and container end, HTML, container start, label, UI page, and duration.</td>
</tr>
<tr>
<td>Tooltip</td>
<td>Tooltip text to display when users point to the variable. Enter a brief note to describe the purpose of the question.</td>
</tr>
</tbody>
</table>
| Example text | Hint that's displayed in the question field before a user enters a value. Applicable for the following variables:  
  - IP address  
  - Email  
  - URL  
  - Single Line Text  
  - Multi Line text  
  - Date  
  - Date/Time |
### Annotation

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show Help</td>
<td>If selected, displays the help text for the variable.</td>
</tr>
<tr>
<td>Always Expanded</td>
<td>This option appears only when the Show help check box is selected.</td>
</tr>
<tr>
<td></td>
<td>If selected, the Help text field value is expanded by default when the catalog item page loads.</td>
</tr>
<tr>
<td>Help tag</td>
<td>If the Always Expanded check box is cleared, click the value specified in this field to display the Help text field value.</td>
</tr>
<tr>
<td>Help text</td>
<td>Help information for a service catalog variable.</td>
</tr>
<tr>
<td>Instructions</td>
<td>Information that requires rich text formatting or adding images to support help information.</td>
</tr>
</tbody>
</table>

**Note:** For HTML tables, use sizes that are within the width of the variable.

### Type Specifications (The fields in this section vary for each variable type)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Width</td>
<td>Width for the variable on the catalog item page, to specify what percentage of the screen size that it can span. This field appears for all variable types except for break, container end, container start, container split, container layout, and label variables.</td>
</tr>
<tr>
<td>Validation (Regex)</td>
<td>Regular expression that validates the user input for the property.</td>
</tr>
<tr>
<td>Validation Error (Regex)</td>
<td>If the expression isn't valid, an error appears.</td>
</tr>
</tbody>
</table>

### Default Value

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>Default value for the variable.</td>
</tr>
<tr>
<td>Variable Attributes</td>
<td>Attributes for this variable.</td>
</tr>
</tbody>
</table>
### Permissions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Roles</td>
<td>Roles that can view the variable before or after requesting the catalog item or record producer. Only a user with the roles specified in this field can view the variable.</td>
</tr>
<tr>
<td>Write Roles</td>
<td>Roles that can edit the variable in the variable editor after requesting the catalog item or record producer. If a user doesn’t have the roles specified in this field, the variable is read-only in the variable editor.</td>
</tr>
<tr>
<td>Create Roles</td>
<td>Roles that can create values for the variable before requesting the catalog item or record producer. If a user doesn’t have the specified role, the variable is read-only before requesting the catalog item or record producer.</td>
</tr>
</tbody>
</table>

### Availability

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visible</td>
<td>If selected, the variable is visible in the item form before ordering the item, in the Variable editor after ordering the item, and in the cart view of the item.</td>
</tr>
<tr>
<td>Read-only</td>
<td>If selected, the variable is read-only in the Cloud User Portal and cannot be edited.</td>
</tr>
</tbody>
</table>

5. **Click Update** once you’ve made all the changes to the variable.

### Manage operation attributes

You can specify attribute values for catalog items help cloud users to correctly provision catalog items. You can configure and manage attributes for provision, pre-provision, and post-provision operations.

Role required: `sn_cmp.cloud_service_designer`.

The attributes for provision, pre-provision, and post-provision operations are generated automatically. You can generate variables for these attributes and also create new attributes for the Blueprint Container Resource operation. You can also use the Manage Attributes screen to create variables for Orchestration attributes.

1. In the Cloud Admin Portal, navigate to **Design > Cloud Catalog Items**.
2. Open a catalog item and click the **Manage Attributes** related link.
   The Manage Attributes page lists the operations.
3. Click an operation to view its corresponding attributes.
4. Configure the attributes for each operation.
   - **Mapping**: View the auto-generated mapping values for each attribute. Mappings specify where to pull the information in the system. To modify the expression mappings, double-
click the mapping value and enter new values. Then click the check mark icon (✓).
Expressions can hold the values for information that's dynamically generated in the system, such as the values for the selections that users make in the Cloud User Portal. See Using expressions in Cloud Management.

- **Generate Variable**: Generates a variable for an attribute. Double-click the value (false), select True, and click the check mark icon (✓).

5. Click **Apply Changes** after you’ve made all the required changes. All the configurations you made to the attributes are saved and you’re returned to the Cloud Catalog Items screen.

**Blueprints**
A blueprint is a specialized catalog item template for offering cloud services, or stacks, to cloud users. Blueprints work with any cloud service provider, such as Amazon AWS Cloud or Microsoft Azure Cloud.

**Blueprint components**
The blueprint is the mechanism that is passed to the cloud provider to establish the cloud assets. Requesting a cloud catalog item is called launching a stack. After the cloud asset is obtained, it can then be discovered, managed, and maintained.
Resource blocks

The cloud resources included in the blueprint (catalog item). For example, a resource block can contain a virtual machine, an app that runs on the VM, and information about the datacenter that hosts the VM. Also referred to as resources.

Connections
The relationship between resources. For example, the VM has a host relationship with the app. There are two types of connections, containment and attached to.

- Containment: a connection where resource block contains another, visually depicted as a top-down relationship as shown in the image. For example, the blueprint container must contain at least one other resource block.
- Attached to: a connection between resources that require binding for direct communication to each other, visually depicted as a side-by-side relationship. For example, the binding between a virtual machine and storage forms an attached to relationship.

Operations

The actions that are performed on resource blocks or on the resource stack to manage the resources. For example, the cloud user can provision, start, and stop a resource.

Resource operations

The steps to perform the operation on the resource. For example, to provision the catalog item, the system must provision the VM and the app that runs on the VM. Click an operation to drill into the resource operations.

Attributes

The details of the resource. Attributes can appear on the catalog item form as catalog properties. For example, the blueprint can provide the version of the app to provision to the VM.

Constraints

An attribute option to constrain a cloud-agnostic resource block, such as a logical datacenter, to be used with a specific cloud provider. For example, a logical datacenter can be constrained to Amazon AWS Cloud. Later, if you switch or add Microsoft Azure Cloud or another cloud provider, you can change the constraint as needed.

Planning blueprints

Before creating blueprints, plan the stacks you want to offer in the service catalog. Prepare the following information before you begin development.

Resource blocks

Plan the resource blocks you want to use in blueprints. Ensure that the appropriate resource blocks are available to use in blueprints, and add them if necessary.

Service offerings (stacks)

To plan a service offering or stack, diagram the resource blocks that it needs, and how they are connected.

Cloud accounts and datacenters

Obtain a list of the cloud accounts and datacenters that the cloud administrator has configured. Determine whether any are constrained.

Service catalog variables and variable sets

Ensure that you have an understanding of how service catalog items use variables and variable sets. Understanding service catalog items and processes help with your understanding of blueprints.
Start here

*Build a Cloud Management blueprint*

Build a Cloud Management blueprint

Use the blueprint designer to create custom blueprints for the Cloud Management application.

Role required: sn_cmp.cloud_service_designer

1. In the Cloud Admin Portal, navigate to **Design > Blueprints**.
2. Click **New**, enter a unique and meaningful **Name** and **Description**, and then specify the service **Category**. Several categories are provided by default.
3. Click **Submit**.

The system creates the blueprint record and displays the Deployment Model tab of the blueprint designer. The Deployment Model tab is a visual canvas where you drop resource blocks. Each resource block represents a CI in a multi-tier application stack. The Deployment Model specifies the resources and connections between resources for the cloud items that you are defining.
4. To add a resource to the deployment model, click **Resources** and then select the resource from the list. The resources appear on the canvas. Move the resource to an appropriate location for the model that you have in mind. Add as many resources as needed.

5. Connect the container resource to the first resource to deploy. To create a connection between resources, select the parent resource, click the connection pad in the resource, and
then drop the connection in any resource with a dotted outline. The system does not allow certain connections (for example, an AWS datacenter cannot host an ARM resource).

6. Add and connect all resources that make up the blueprint. You must add and connect all resources before you configure operations and attributes in the Operations section.

As you update the resources in the Deployment Model tab, the system generates the Cloud Template: a complete representation of the blueprint in YAML script. You can export the Cloud Template representation of the deployment model and later import it into the Cloud Management application on another instance.

7. Configure the operations that the system performs on the resources and the attributes to use while performing the operations. By default, Provision, Stop, Start and Deprovision operations are provided. You can either select an existing operation or create a new blueprint operation.

- Select an existing operation: In the Operations section, select the Provision operation check box and click Provision to view a graph. You can rearrange the graph by deleting and adding operations. You can click Optimize All Plans for the Cloud Management application to suggest an optimized plan for you. The Cloud Management application looks at the blueprint you created and based on the attributes, the mappings, and the dependencies of each resource, suggests an optimized plan to execute. All the attributes for the Provision operation appear in the window. You can select or de-select any attribute. Once you configured the Provision operation, you can select the next operation and configure its attributes as needed.

- Create a new blueprint operation: Click New in the Operations section. The New Blueprint Operation dialog box appears. Enter a name, select an operation type, enter input parameters for the new operation, and click Save. The new blueprint operation appears in the Operations section.

8. If you have completed the design of the blueprint and you do not want to configure the blueprint any further, click the Draft toggle button to publish the blueprint. A message appears saying that your blueprint has been published. To continue configuring the blueprint, then skip this step.

9. To configure parameters such as variable sets, rules, life cycle operations for the blueprint catalog item, click the Catalog tab and then click the catalog. The Cloud Catalog Item screen appears.

10. Click Update if you made any changes to the form.

11. Click Publish to publish the blueprint. To view all published catalog items, click Published in the Blueprints landing page.

Execute blueprint operations in parallel
You can execute your blueprint resource operations to run in a linear or in a parallel way.

Role required: sn.cmp.cloud_service_designer.

By default, the blueprint resource operations are listed serially and follow an order. You can run operations in parallel by arranging them in branches. Identify the operations specified in a blueprint that have no dependencies and can run in parallel. Arrange such operations in a single branch. The Blueprint Designer runs these operations together. If an error occurs in a branch, the operations in the affected branch cease to execute. However, the operations in the other branches, that are unaffected by the error, continue to execute. The parallel way of executing operations hence cuts down time in provisioning stacks.

1. In the Cloud Admin Portal, navigate to Design > Blueprints.

2. Click a blueprint, click the Deployment Model tab and then click Operations. All the operations for the blueprint are listed.

3. Click an operation to view a graph. You can rearrange the graph by deleting and adding operations.
4. Click **Optimize Plan** for the Cloud Management application to suggest an optimized plan for you. The Cloud Management application looks at the blueprint you created and based on the attributes, the mappings, and the dependencies of each resource, the application suggests an optimized plan to execute.

Add a resource operation to the execution plan

The base system includes a large set of predefined resource operations. Rarely, you might add a custom operation for a resource.

**Role required:** sn_cmp.cloud_service_designer

1. While working in the blueprint designer, click the **Operations** tab and select the appropriate blueprint operation. As an example, you may select the **Provision** operation.

2. Click **Add**.

   The Add Resource Operation dialog box appears.

3. Fill in the form fields (as shown in the table).

<table>
<thead>
<tr>
<th>Resource</th>
<th>Select a resource to perform the operation on.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Select an interface for the resource.</td>
</tr>
<tr>
<td>Operation</td>
<td>Select an operation to perform for the step. By default, <strong>Provision</strong>, <strong>Stop</strong>, <strong>Start</strong> and <strong>Deprovision</strong> are provided. You at least need a provision operation for a user to order a cloud catalog item.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

   The new operation is added to the operation graph. The system adds the appropriate parameters for the operation.

5. Click the **Create Form Parameter** check box to select a parameter. Parameters differ for each operation. Enter a mapping expression for the selected parameter in the **Mapping** field. See [Using expressions in Cloud Management](#).

6. Click **Save**.

**Blueprint attributes**

When you add a blueprint operation to a resource, the system adds the appropriate attributes. You can choose whether or not to make the attributes available as catalog properties on the form when the system performs the operation.

**Common attributes for the provisioning operation for virtual machines**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VmFolderName</td>
<td>The name of the virtual machine folder.</td>
</tr>
<tr>
<td>ResourceGroupName</td>
<td>The name of the resource group to which the resource belongs.</td>
</tr>
<tr>
<td>StackName</td>
<td>The name of the stack that the user is provisioning.</td>
</tr>
<tr>
<td>CloudAccount</td>
<td>The cloud account to which the resource stack belongs.</td>
</tr>
<tr>
<td>OSProfile</td>
<td>The resource profile, in this case the Operating System profile, upon which the cloud resource is based.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DiscoverAndAttachIP</td>
<td>Finds a free elastic IP and assigns it to a virtual machine on AWS. By default, if a VM is deployed on a VPC in AWS, it will not get a public IP. If elastic IPs are not available, no IP can be assigned to the VM.</td>
</tr>
</tbody>
</table>

**Warning:** It is recommended that you do not use **KeyPreference**. If you use the attribute, do not set **KeyPreference** to **user**.

Configure a blueprint rule

Configure a rule that contains actions the system can take on blueprint attributes.

Role required: sn_cmp.cloud_service_designer

1. Open a blueprint that has the attribute for which you want to create a rule.
2. On the **Catalog** tab, click the operation, such as the provision operation, that contains the attribute.
3. In the Variable Sets related list on the Cloud Catalog Item form, click the name of the variable set that contains the variable.
4. In the Cloud Variables related list on the Variable Set form, click the name of the variable for which you want to create the rule.
5. In the Rules related list, click **New** and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name and description.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a number that tells the system which record to evaluate first in relation to others. The record with the lowest number runs first.</td>
</tr>
</tbody>
</table>

6. Right-click the form header and select **Save**.
This example rule is based on the **CloudAccount** property and contains an action that tells the system to update the list of locations when the user changes the cloud account.

**Example blueprint rule**

Configure an optional *condition* and a mandatory *action*.

Configure a blueprint rule condition
- Configure the condition on a blueprint form event rule that determines when the actions in the rule can run. Conditions are optional.

Role required: `sn_cmp.cloud_service_designer`

1. Open a *rule for a variable* on a blueprint.
2. On the Conditions related list, click **New**.
3. Fill out the form fields (see table).
## Field Description

**Operand**
Enter the property against which you can perform the conditional evaluation. For example, if your blueprint has a property which holds the value of a datacenter, enter the property name here.

**Operator**
Select an operator that evaluates the operand.

**Value**
Enter a value that the evaluated event rule must match.

**Condition Type**
If you configure multiple conditions, select the logical operator that determines how the system evaluates the condition in relation to the next condition (the next condition is determined by the **Order** field):

- **AND**: this condition must meet the criteria before the policy is eligible for execution.
- **OR**: this condition or the next condition that matches the criteria can allow the policy to be eligible for execution.

If you do not select an option, **AND** is used by default.

**Order**
Enter a number that tells the system which record to evaluate first in relation to others. The record with the lowest number runs first.

**Status**
Keep the status active to have the system process the condition. Change the status to inactive when you do not want the system to process the condition.

### 4. Click **Submit**.

This example condition applies when the user selects the AWS Datacenter for the Logical_Datacenter property, which is a field on the catalog item form:

![Example blueprint condition](image)

**Configure a blueprint action.**

Configure actions that the system runs when blueprint form attribute actions or event actions are triggered. You can configure actions like making a field mandatory, showing or hiding a field, or automatically populating a field.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Role required: sn_cmp.cloud_service_designer

1. Open a rule for a property or an event on a blueprint.
2. On the **Actions** related list, click **New** and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique and descriptive name.</td>
</tr>
<tr>
<td>Action Type</td>
<td>Select an action type:</td>
</tr>
<tr>
<td></td>
<td>▪ Reload:</td>
</tr>
<tr>
<td></td>
<td>▪ Show:</td>
</tr>
<tr>
<td></td>
<td>▪ Hide:</td>
</tr>
<tr>
<td></td>
<td>▪ MakeMandatory:</td>
</tr>
<tr>
<td></td>
<td>▪ MakeNonMandatory:</td>
</tr>
<tr>
<td></td>
<td>▪ TriggerChangeEvent:</td>
</tr>
<tr>
<td></td>
<td>▪ SetValue:</td>
</tr>
<tr>
<td>Datasource</td>
<td>Select an action type:</td>
</tr>
<tr>
<td></td>
<td>▪ Reload Cloud Account:</td>
</tr>
<tr>
<td></td>
<td>▪ Reload Cloud Location:</td>
</tr>
<tr>
<td></td>
<td>▪ Check Quota:</td>
</tr>
<tr>
<td></td>
<td>▪ Reload Resource Profile:</td>
</tr>
<tr>
<td></td>
<td>▪ Reload From Profile:</td>
</tr>
<tr>
<td>Target Field</td>
<td>Enter the name of the field on the form that the action applies to. For example, if you selected <strong>TriggerChangeEvent</strong> for the action type, the value of the field changes when something changes on the form, as specified in the <a href="#">blueprint rule</a>.</td>
</tr>
<tr>
<td>Value</td>
<td>Enter an optional value that you want to assign to the target field. For example, if you selected <strong>SetValue</strong> for the <strong>Action Type</strong> field and you want to update the <strong>Resource Group Name</strong> field, you can use the value ${ResourceGroup}, which pulls the names of the resource groups from the blueprint.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter a number that tells the system which record to evaluate first in relation to others. The record with the lowest number runs first.</td>
</tr>
</tbody>
</table>

3. Right-click the header and select **Save**.
4. Enter an optional **Key** and **Value** in the **KeyValues** related list. The value in the **Value** field is applied to the attribute in the **Key** field.
5. Create as many actions as necessary.

This example action applies to the [blueprint rule example](#), which is based on the CloudAccount attribute. The action tells the system to update the list of locations when the user changes the cloud account.
Example blueprint action

*Publish a blueprint as a cloud catalog item.*

Create a blueprint for configuration management provider integration

After you set up your configuration management provider and run Discovery on it, configure resource blocks.

Role required: sn_cmp.cloud_service_designer

For an example of a virtual machine configured to work with Ansible, see *Create a blueprint for Ansible Tower integration.*

1. In the Cloud Admin Portal, navigate to **Design > Blueprints.**
2. Click **New.**
3. Create a deployment model with a container, virtual machine, and a datacenter.
4. Click the **Operations** tab, select an operation, and then click **Add**.
   The Add Resource Operation dialog box appears.

![Add Resource Operation](image)

**Add Resource Operation**

- **Resource**: Virtual Server
- **Operation**: Bootstrap Node

5. Add the following resource operations for the Virtual Server one at a time:
   - **Bootstrap Node**: prepares the virtual server to communicate with Ansible Tower.
   - **Register Node**: registers the configuration management provider.
   - **ExecuteConfigPackages**: runs the applications or playbooks.

6. Click **Submit** once you have added the resource operations.
   The operations are added to the graph.
7. Make the following changes to these operations, and then click **Save** after each change:

<table>
<thead>
<tr>
<th>Operation</th>
<th>Make this change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blueprint Container Resource. Provision</td>
<td>Enable the <strong>WorkloadConfigProvider</strong> and <strong>WorkloadConfigProvider Type</strong> parameters. On the Cloud User Portal, users can see these fields to select the configuration management provider.</td>
</tr>
<tr>
<td>Management Attributes</td>
<td>Every operation has management attributes.</td>
</tr>
</tbody>
</table>

8. Go to the Cloud User Portal. Select the workload provider type in the order catalog form. Based on the workload provider type you select, the **Workload Config Provider** field gets loaded.

9. Select an application profile. Based on the profile, the configuration overrides appear. These overrides are the discovered config installable parameters.

10. Publish the catalog item.

---

**Form configuration with blueprints**

Blueprints control what the user sees on the catalog order form. You can configure the form to show or hide fields, populate fields with default values, and otherwise create a unique form for the user based on criteria.

You can also change the values in form fields using policies. See *Create a cloud policy* and *Create a policy action script* for examples.

---

**Components of blueprint form configuration**

Blueprints use these components to control how the system uses catalog form fields:

- **Catalog variables** define the basic structure of each field on catalog order forms.
- **Rules** specify an action that the system takes on a field when a condition is met.
- **Scripts** query the system for information that you need to show the catalog requester, populate form fields, and process the data that users submit.
UI groups to put related fields together in a tab. Users can progress through tabs to fill out all the form fields.

Start here

These tasks show you the basics of form configuration with blueprints:

- Add fields to a cloud catalog item order form
- Control visibility default values for Cloud catalog items
- Create a variable set for Cloud Management

Examples

These examples guide you through form configuration based on a specific goal: Cloud catalog form configuration examples.

Add fields to a cloud catalog item order form

Add fields to a cloud catalog form if you want the user to enter additional information beyond what a default blueprint provides.

- You need a blueprint that you published to the catalog. The blueprint should be in the draft state. See Build a Cloud Management blueprint if you did not complete this prerequisite.
- By default, cloud items appear in the Cloud Catalog, rather than the standard service catalog that comes with your instance. However, Blueprints utilize several features from the standard service catalog. It is suggested that you familiarize yourself with how the service catalog works. See Service Catalog to get started.
- Role required: sn_cmp.cloud_service_designer

Fields are represented as variables in blueprints. To add a field, you add a variable to an operation in the blueprint.

1. Open a blueprint that you already created.
2. Click the Catalog tab, and then click the operation, such as Provision, that you want to add the variable to.

The Cloud Catalog Item form opens.
3. In the Variable Sets related list at the bottom, click the set in which you want the variable to belong. For example, click General Info, which is a default variable set for all blueprints, to have the field appear in the first tab on the catalog order form. The Variable Set form opens.

4. In the Cloud Variables related list at the bottom, click **New**.

5. Fill in the important form fields, such as:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Select the type of variable, such as <strong>Single Line Text</strong> for a text field, <strong>CheckBox</strong>, or <strong>Date</strong>, for example.</td>
</tr>
<tr>
<td>Question tab</td>
<td>In <strong>Question</strong>, which is the value the user sees for the field on the catalog order form, enter a name. The <strong>Name</strong> field, which is the actual name of the variable, is automatically populated with the same value in lower case.</td>
</tr>
<tr>
<td>Other tabs</td>
<td>Fill in other fields on the remaining tabs to control the field’s visibility, form help text, default value, and access rights.</td>
</tr>
</tbody>
</table>

**Note:** This variable will not be available for orchestration. If you intend to use this variable for orchestration, please add it through Cloud Template or Blueprint instead.
6. Click **Submit**.

   The Variable Set form opens, showing the new variable in the Cloud Variables related list.

7. To control the position of the field on the form, change the value in the **Order** column. A lower order value moves the field higher up on the form, and vice versa.
In this example, the order is set to 1, which means the Email address field appears after the Logical Datacenter field and before the Cloud Account field.

8. On the form banner in the upper-left, click **Try It** to preview the order form as it would look to a user from the Cloud Catalog.

The form appears with the new field that you created.
9. Click the back arrow on the upper-left of the form.
10. Toggle the catalog item to Publish.

Control visibility default values for Cloud catalog items
Through blueprints, you can control how form fields appear to users in the cloud catalog. For example, you can control where the fields appear to users on the catalog item form, and whether they are read only or mandatory.

1. Open the blueprint to modify, and then toggle it from Published to Draft.
2. Click the Catalog tab.
   The Request Operation sub-tab opens and displays the published catalog items for the Provision operation.
3. Click the Provision tile for the catalog item.
   The settings in the Provision operation control the form that the user sees upon initial request of the item, before it is provisioned.

The Cloud Catalog Item form opens.
4. In the Variable Sets related list at the bottom, click the set in which you want the variable to belong. For example, click General Info, which is a default variable set for all blueprints, to have the field appear in the first tab on the catalog order form. The Variable Set form opens.

5. Click the variable you want to modify. The Cloud Variable form opens.

6. Complete any of the following actions.

<table>
<thead>
<tr>
<th>To</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make the field mandatory</td>
<td>Select the Mandatory check box. Clear the check box to make the field optional.</td>
</tr>
<tr>
<td>Make the field read-only</td>
<td>Click the Availability tab, and then select the Read Only check box. Clear the check box to make the field editable. Use this option to automatically populate a value in the field that the user cannot change.</td>
</tr>
<tr>
<td>Show or hide the field</td>
<td>Click the Availability tab, and then select or clear the Visible check box.</td>
</tr>
<tr>
<td>Move the field to another tab</td>
<td>Select a different Variable set. General Info and Provision are provided in the base system, and you can configure additional form UI groups. To add a new tab for organizing fields in the catalog item, see Create a variable set for Cloud Management.</td>
</tr>
<tr>
<td>Set a default value</td>
<td>Click the Default Value tab and enter the value in Default value. If the field is not read-only, users can change the value on the cloud catalog form. Reference fields must use the Sys ID of the record that represents the default value.</td>
</tr>
</tbody>
</table>
| Add help for the user | Do any of the following:  
- Click the Question tab, and add a value in the Tooltip field to show a bubble with the text when the user hovers the cursor over the field.  
- Click the Annotation tab, select Show help, and then enter values in the Help text or Instructions fields to display additional text to the user above the field. |

7. Click Update.

8. Toggle the catalog item to Publish.

In this example, the Business Service field:

- Is populated with the Email business service record from the cmdb_ci_service table based on the Sys ID of the record.
- Is read-only so users cannot change it.
- Is mandatory because it is considered necessary before the system can provision the virtual resource.
- Contains help text to tell the user that the default selection is Email.
Cloud catalog form configuration examples

Several examples are provided to help you understand how to manipulate Cloud catalog order forms. These example cover blueprint rules, blueprint variables, and blueprint form UI groups.

<table>
<thead>
<tr>
<th>Example</th>
<th>Goal</th>
<th>Blueprint components used</th>
</tr>
</thead>
</table>
| **Populate the location based on the cloud account** | Use existing blueprint rules to illustrate how the Location field is populated based on the Cloud Account that the user selects. | This example illustrates:  
  • Rules |
| **Remove a field from a cloud catalog form** | Hide the Cost Center field on the catalog form when the catalog requester belongs to the Database group. | This example illustrates:  
  • Rules  
  • Resource pools |
<table>
<thead>
<tr>
<th>Example</th>
<th>Goal</th>
<th>Blueprint components used</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Populate stack name with a script</strong></td>
<td>Give the catalog requester a predefined stack name that includes the ID of the catalog requester. The field remains editable to the user can change it if necessary.</td>
<td>This example illustrates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Rules</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Resource pools</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Scripted values</td>
</tr>
<tr>
<td><strong>Add fields to a cloud catalog item order form</strong></td>
<td>Create a catalog variable and property, and use it to publish a new filed on a form.</td>
<td>This example illustrates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Catalog variables</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Field order</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Rules</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Resource pools</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Scripted values</td>
</tr>
<tr>
<td><strong>Set default values for a cloud catalog form fields</strong></td>
<td>Set a default value on a property and on a blueprint rule.</td>
<td>This example illustrates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Default variables</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Rules</td>
</tr>
<tr>
<td><strong>Add a UI group with additional fields</strong></td>
<td>Group catalog form fields into tabs to better organize the form and guide the user from one set of settings to another.</td>
<td>This example illustrates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UI groups (tabs)</td>
</tr>
</tbody>
</table>

Populate the location based on the cloud account

This example walks you through an existing set of blueprint rules that Cloud Management provides by default.

Before you can follow this example, you should be familiar with creating blueprints and blueprint rules. See these resources:

- **Blueprints**
- **Build a Cloud Management blueprint**
- **Configure a blueprint rule**

Role required: sn_cmp.cloud_service_designer

The goal of this example is to show you how a default set of blueprint rules accomplishes the task of populating the Location field based on the Cloud Account. Because this example walks you through default data, you do not need to create any records if you are following these steps.

This blueprint, which includes a virtual server that can be provisioned in an AWS datacenter, is used to illustrate the example:
1. Open a similar blueprint with a virtual server or similar.
2. On the blueprint form, click the Catalog tab, and then click the tile that represents the Provision operation.

3. Click the General Info variable set in the Variable Sets related list.
4. Click the CloudAccount variable to open the Cloud Variable form.

Notice these items:
- In the Rules related list at the bottom, two rules are provided:
  - ReloadLocation reloads the value in the relocation field. This rule runs first because it has a lower order number than the other rule.
  - TriggerChangeEvent Location tells the system to trigger the actions for the rules that are already configured for the Location variable.
- The Lookup Select Box field specifies that the Cloud Account field provides a selection for the user to make. The options that the user can select comes from the CloudAccountPool, which is on the Type Specifications field.
5. Click the **Reload Location** rule, and then click the **Reload Location** action from the Actions related list.

Notice these items:
- The **Action Type** is **Reload**, which tells the system to reload the catalog item form so that a new value for the field specified in the **Target Variable (the Location)** can be populated.

![Action Reload Location](image)

- The **DataSource** field tells the system to reload the location.
- The **KeyValues** list, which is only used when you specify a **Reload** action type, specifies that the user is passing in the cloud account via the form. The cloud account is necessary to reload the correct value for the location because the location is loaded from a pool. That pool has a filter that requires the input of the cloud account to return the correct location options.

6. Navigate back to the CloudAccount catalog property form and open the other rule: **TriggerChangeEvent Location**.

7. Click the **Actions** related list, open the **TriggerChangeEvent Location** action, and then click the **TriggerChangeEvent Location** action from the Actions related list.

Notice these items:
- The **Action Type** is **TriggerChangeEvent**, which tells the system to trigger the actions for the rules that are already configured for the Location property. By default, several rules are
already configured for the Location property. These include rules that reload other fields, like **OS Profile**, **Network**, and so on.

- No **KeyValues** are required because this action is not a reload action.

The result of these two rules on the CloudAccount attribute is illustrated in the following example:

8. Perform these steps to see why the system populated the value **AWS Datacenter - us-east-1** in the **Location** field and how the system determines which location is related to the **My Cloud Account** value:
a) In the same blueprint, open the **General Info** variable set, and then open the **Location** parameter.

b) Click the **Type Specifications** tab.

Notice the values that can be populated for the **Location** field from the **CloudLocationPool** resource pool.

c) Navigate to **Manage > Resource Pools**, open **CloudLocationPool**, and notice these items:

- The locations come from the records in the **Lookup Table**, which in this example is **Associated Datacenter (sn_cmp_ca_assc_datacenter)**.
- The actual value to display to users in the Location field on the catalog item form comes from the **Lookup Field**. In this example, the **Lookup Field** specifies the Alias column from the Associated Datacenter (sn_cmp_ca_assc_datacenter) table. The **Alias** column holds the actual, user-readable names for all locations.

d) Go to the standard interface of your instance, not the Admin Portal, and enter **sn_cmp_ca_assc_datacenter.list** in the navigation filter.

The list of datacenters that are already associated with cloud accounts appears.
Notice that the **My Cloud Account** cloud account is already associated with **AWS Datacenter - us-east-1**, which is the value that the user sees in the Location field. The system knew to display that value because of the association in this Associated Datacenter table. The system created this association when your Cloud Management administrator created the cloud account for AWS, and then ran discovery on the us-east-1 datacenter in that cloud account.

It is possible that multiple datacenters could be associated with **My Cloud Account**. In that case, the Location field reloads all the associated datacenters and in the Location field. Since the Location field is of type Select, the field allows the user to select which location to choose.

### Remove a field from a cloud catalog form

This example illustrates how to remove a field from the Cloud Management catalog form when the user belongs to a specific group.

Role required: sn_cmp.cloud_service_designer

Before you can follow this example, you should be familiar with creating blueprints and blueprint rules. See these resources:

- [Blueprints](#)
- [Build a Cloud Management blueprint](#)
- [Configure a blueprint rule](#)

The goal of this example is to hide the Cost Center field on the catalog form when the catalog requester belongs to the Database group. You can follow an example like this one when you want to show or hide catalog item fields based on values in other fields.

1. Create a blueprint with a virtual server. For this example, a blueprint with a virtual server on an AWS datacenter is used.

2. On the blueprint form, click the **Catalog** tab, and then click the tile that represents the **Provision** operation.
3. Click the General Info variable set in the Variable Sets related list.
4. Click the CostCenter variable to open the Cloud Variable form.
5. Configure the rule that removes the cost center based on the user group:
   a) In the Rules related list, click New.
   b) Fill out the Name and description Fields.
   c) Right-click the form header and select Save.

6. Configure the condition that requires the User Group to equal Database:
   a) In the Conditions related list, click New.
   b) Fill out the form fields as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value to enter or select</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operand</td>
<td>UserGroup</td>
<td>This value is the User Group field that the user selects.</td>
</tr>
<tr>
<td>Operator</td>
<td>--None--</td>
<td>Keep this value --None-- because this condition is the only condition you need to configure. If you add more conditions, you can select another operator to determine how the system evaluates this condition in relation to the others.</td>
</tr>
<tr>
<td>Value</td>
<td>287ee6fed9fe198100ada7952</td>
<td>This value is the Sys ID of the Database user group, which is provided as part of the demo data in your instance. See Sys ID for more information.</td>
</tr>
</tbody>
</table>

   c) Click Submit.

7. To understand why the Sys ID of the user group is required for the system, perform these steps:
   a) Open the blueprint again and click the Provision operation again.
   b) Click the General Info from the Variable Sets related list.
   c) Click the UserGroup variable.

   Notice that on the Type Specifications tab, the system populates user groups from UserGroupPool. The system runs the scripted pool filter named GroupsUserBelongsTo.
d) In the Cloud Admin Portal, navigate to Manage > Resource Pools.

e) Find and open UserGroupPool.

Notice that the values for the pool are derived from the table in the Lookup Table field, which in this case is the Group (sys_user_group) table.

Also notice that there are two queries in the Resource Pool Filters related list:

- **All** is a generic query that returns the list of all records from the Lookup Table. The value it returns from that table is from the Lookup Field, which in this case is the Sys ID. However, the system ignores the All query in this case because the scripted query is present.

- **GroupsUserBelongsTo** is a scripted query that returns the Sys ID of records in the Lookup Table.

**Important:** When a scripted query is present, the system uses it instead of the generic ALL query.

Both the generic query and the scripted query return the Sys ID of the record in the lookup table. But what is actually happening here is that the system ignores the All query and the value specified in the Lookup Field, and it uses the GroupsUserBelongsTo script.
A scripted query

Populate stack name with a script

This example illustrates how to populate the name of the stack that a user requests from the Cloud Management catalog by utilizing the user's ID. The example uses a script include that you can call from a blueprint rule action.

Role required: sn_cmp.cloud_service_designer

Before you can follow this example, you should be familiar with creating blueprints and blueprint rules. See these resources:
· **Blueprints**
· **Build a Cloud Management blueprint**
· **Configure a blueprint rule**

This example uses a script include. If you want to create similar script includes, familiarize yourself with the Glide System APIs that are part of your instance. See the [Developer site](#) for a comprehensive list of all APIs. See [Script Includes](#) for more information about how the system uses these scripts.

The goal of this example is to take the ID of the catalog requester and populate it in the Stack Name field. The field remains editable. Follow an example like this one when you want to loosely enforce a naming or labelling convention on certain values, but still allow the user to enter the text that they want.

1. Create a script include that returns the user ID of the person requesting the catalog item:
   a) In the standard interface for the instance (not in the Cloud Admin Portal), click the gear icon (⚙️) in the upper left, and then click **Developer**.
   b) In the **Application** field, select **Cloud Management Platform**.

The application scope of your instance changes to **Cloud Management Platform**. This selection means that the Cloud Management application can find the script include.

c) Close the System Settings window.

d) Navigate to **System Definition > Script Includes**.
e) Click **New**.

f) Fill in the form fields (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a descriptive name without spaces.</td>
</tr>
<tr>
<td>Accessible from</td>
<td>Select <strong>All application scopes</strong>.</td>
</tr>
<tr>
<td>Script</td>
<td>Enter this script:</td>
</tr>
</tbody>
</table>

```javascript
var SetFields = Class.create();
SetFields.prototype = {
    initialize: function() {
    },
    setStackName: function(){
        return gs.getUserName();
    },
    type: 'HideFormAttributes'
};
```

The function `setStackName()` returns the user name of the catalog item requester. You can call this function from a blueprint rule action.
g) Click **Submit**.

h) Click the gear icon (🔧) in the upper left again, and then change the Application back to **Global**.

2. In the Cloud Admin portal, navigate to **Design > Blueprints** and open the relevant blueprint.

3. On the blueprint form, click the **Catalog** tab, and then click the tile that represents the **Provision** operation.
4. Click the **Stack Name** variable set in the Variable Sets related list. Notice that Stack Name is a single line text field that takes text input.

5. Create a rule to populate the stack name:
   a) Scroll down to the bottom of the form and click **New** in the **Rules** related list.
   b) Provide a name and description.
   c) Click **Submit**.

6. Configure the action as follows:
   a) From the Rule form, scroll down and click **New** in the **Action** related list.
   b) Configure the rule as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Provide a meaningful name and description, such as Set Stack Name to User ID.</td>
</tr>
<tr>
<td>Action Type</td>
<td>Select SetValueByScript to tell the system to call the script include that you specify in the Value field.</td>
</tr>
<tr>
<td>Target Variable</td>
<td>Select StackName to tell the system to set the value of the Stack Name field.</td>
</tr>
</tbody>
</table>
| Value           | Call the method in the script include that returns the user ID. The call must be in the following syntax:  
                  \$\{Script:ScriptIncludeName.methodName[]\}  
                  In this example, use  
                  \$\{Script:SetFields.setStackName[]\}  |
c) Click **Submit**.

In the Cloud User Portal, the catalog order form for the Provision operation shows the user ID of the logged-in user. This example uses the user Able Tutor, which comes in the base system as demo data. The value in the **Stack Name** field is the same value as in the User ID field on the User (sys_user) table. The profile icon for the same logged in user is also displayed in the upper right of the Cloud User Portal.
Set default values for a cloud catalog form fields
You can set default values for all users or you can populate fields with default values based on conditions. This example shows you how to set a default schedule profile and business service.

Role required: sn_cmp.cloud_service_designer

Before you can follow this example, you should be familiar with creating blueprints and blueprint rules. See these resources:

- [Blueprints](#)
- [Build a Cloud Management blueprint](#)
- [Configure a blueprint rule](#)

The goal of this example is to display a default value for both the schedule profile and the business service field. Each field derives its values differently:
- The schedule profile derives its values from a pool filter that returns a text value. Therefore, you must set the default value by applying the correct, matching text for one of the schedule profile options that appear to the user in a list.
- The business service derives its values from a reference to the Service (cmdb_ci_service) table. Therefore, you must set the value by specifying the Sys ID of the record that you want to display.

The example in this topic sets the values in different ways:
- The schedule profile default value is set directly on the Schedule Profile property and applies to all users without any conditions. You might want to do this to limit the use of virtual resources to specific schedule, such as during business hours only, for all users.
- The business service default value is set using an action on a blueprint rule. You might want to use a rule to set a default value when you want the value to be the default only when certain conditions are met. For example, if the user belongs to a user group that deals with only one business service, set the value for the user so they do not have to sort through a long list of all the business services in your organization.

1. Create a blueprint with a virtual server. For this example, a blueprint with a virtual server on an AWS datacenter is used.

2. On the blueprint form, click the Catalog tab, and then click the tile that represents the Provision operation.

3. Click the General Info variable set in the Variable Sets related list.
4. Click the ScheduleProfile variable.
Notice that the **Pool** field on the **Type Specifications** tab specifies the pool from which the values for the schedule profile are derived: ScheduleProfilePool.

The **ScheduleProfilePool** pool specifies the **All** filter, which returns the values in the Name field from the Schedule Profile (sn_cmp_schedule_profile) table. These values are static text values, which means that you can specify the exact text value for the name of the schedule profile that you want to make the default.

5. **On the Default Value** tab of the ScheduleProfile property form, enter the text that matches the value in the Name field from the Schedule Profile (sn_cmp_schedule_profile) table. In this example, that value is **8-5 weekdays**.
6. Click **Update**.
7. Click the **BusinessService** variable.

Notice that the **Reference** field on the **Type Specifications** tab specifies the Service (cmdb_ci_service) table, from which the values for the business service are derived.

By default, to refer to a specific record in a table, you must know the Sys ID of the record.

8. To obtain the Sys ID of a business service:
   a) In the main interface for the instance (not the Cloud Admin Portal interface), enter `cmdb_ci_service.list` in the application filter.
   b) Find the business service that you want to make the default.
   c) Right-click the record, and then select **Copy sys_id**.

   This example uses **Workday Enterprise Services**.
9. On the BusinessService variable form, create a rule, action, and condition with these values:
   a) In the Rule related list, click New.
   b) Provide a name and a description.
   c) Right-click the form header and select Save.
   d) In the Action related list for the rule, click New.
   e) Provide a name, and then configure these fields as specified:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Type</td>
<td>SetValue</td>
</tr>
<tr>
<td>Value</td>
<td>The Sys ID of the business service that you copied.</td>
</tr>
</tbody>
</table>

In this example, the business service is set to Workday Enterprise Services by default, which is represented in the system with the Sys ID that you already copied.
<table>
<thead>
<tr>
<th>Name</th>
<th>Set Business Service Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display name</td>
<td>Set Business Service Action</td>
</tr>
<tr>
<td>Action Type</td>
<td>SetValue</td>
</tr>
<tr>
<td>Target Variable</td>
<td>BusinessService</td>
</tr>
<tr>
<td>Value</td>
<td>6a78f28f933a31003b4bb095e57ff8b8a</td>
</tr>
</tbody>
</table>

f) Click **Submit**.
You can create a condition or multiple conditions that must be met before the system sets the default value for the business service. For example, you might want to change the default value of business service to Workday Enterprise Services only when the group that the user belongs to is the Workday Admins group.

Resource blocks in Cloud Management

Resource blocks act as a type of middleware component between blueprints, the Cloud API (CAPI), and the CMDB.

The following graphic illustrates how resource blocks fit between other components in the system:
Blueprints

Each resource block becomes a building block within a blueprint. The blueprint eventually becomes a catalog item (also called a stack) that a user provisions from the Cloud Catalog.

Cloud API (CAPI)

Each resource block defines allowable operations, such as Provision and Deprovision. These operations call various components within CAPI to carry out the operation that is received from the stack.

The CMDB

Each resource block is based on a CI type from the CMDB. Resource blocks also have a response processor that handles responses from the cloud provider to take some type of action in the CMDB, such as creating or updating a CI.

How resource blocks are connected

Resource blocks are connected to each other in blueprints. For example, in this blueprint, a virtual server resource block is connected to storage. The virtual server and the storage are both connected to an AWS datacenter resource block.
For these connections to be successful, each resource block needs to specify a type of interface. This graphic illustrates these interfaces:
Types of interfaces for resource blocks

- The guest interface connects to the resource that is above it. The guest interface contains the operations (also called operation signatures), which allows users to take action on their resource.
- The host interface connects to the resource below the resource block in the blueprint. The host interface also contains operation signatures.
- The bindings connect adjacent resources.

Resource blocks can have as many guest interfaces, host interfaces, and bindings as necessary. Connections can be made in different directions:

- Vertical connections require the top resource block to have the same host interface as the bottom resource block guest interface.

  **Note:** In blueprints, this vertical connection is called a containment connection.

- Horizontal connections bind one resource to another, but both resources have to reside on top of the same resource block.

  **Note:** In blueprints, this horizontal connection is called an attached-to connection.

The same example blueprint with a virtual server with attached storage on an AWS datacenter, the resource blocks are configured as this graphic illustrates:
The binding between the virtual server and the storage is unidirectional. You configure the binding on the virtual storage resource block only by specifying the storage resource block. You do not need to configure another binding on the storage resource block.
Guest interfaces and operations

Each guest interface provides a default set of operations (also called operation signatures) that allow the user to choose what they want to do with a virtual resource. The most common operation is Provision, which means that a virtual resource is created. Many guest interfaces and corresponding operations are provided by default with the Cloud Management application. You can use these default guest interfaces for many of your resource blocks.

If you do need to create a new guest interface, these operations are provided by default:
- Start
- Stop
- Provision
- Deprovision
- Cleanup
- ModifySchedule
- ModifyLease

Resource block component details

Each resource block contains these components:

Layers

The logical layer in the stack that this resource block belongs to. The layers are provided by Cloud Management by default. When you configure resource blocks, you can choose the one that relates to you.

Operations for each guest interface

As mentioned, each guest interface provides operations. For example, the most common operations for a virtual server are Provision, Deprovision, Start, Stop.

Input parameters for each operation

The input parameters hold values that the cloud provider uses, via CAPI, when operations are run. These parameters hold the information that the user selects while performing an operation, such as provisioning, on a stack.

For example, a user typically must specify the location, which is a datacenter, that a stack belongs in. The value that the user selects in the Location field is held in the Location parameter. The system takes this parameter, which is available by default for the virtual server resource block, and passes it to the CreateNode CAPI interface operation. The CreateNode interface operation tells the cloud provider to create (or provision) the actual virtual server.

Steps for each operation

Each operation requires a step to do the actual work. Each step calls a specific interface and method in CAPI, and passes along the input parameters required by that method. For example, the Provision operation on a virtual server provides one step to first connect and create a virtual machine, and another step to create the node in the cloud provider. Each step provides different input parameters. To connect and create a virtual machine, the cloud account, among many other parameters, is required. To create the node, the cloud account is not required.

You can add as many steps as needed for each operation. You can also use output attributes to pass values in attributes from one step to another step, or from steps in one operation to
steps in a subsequent operation. For example, the **Provision** operation for a virtual server can provide an output attribute for the network interface ID and another for the node ID. When the cloud provider sends the response, the actual ID of the network and node are parsed out of the response and saved in the output attributes. Subsequent steps can then use the IDs to perform actions.

**Response processors**

Response processors tell the system what to do in the CMDB. When the cloud provider sends a response after an operation is run, the processor handles the response.

*i Note:* These processors are scripts that are system includes. They are not MID Server script includes like the CAPI scripts that you can create to integrate with the cloud provider.
How components work together

Users can see catalog items, also called stacks, in the Cloud Catalog. A catalog item is based on a blueprint, which in turn is composed of resource blocks that are connected together.

1. The user kicks off the process by requesting a stack. The request is the **Provision** operation in the system.
2. The system runs the steps in order for the **Provision** operation for the resource blocks that comprise the stack.
3. Each step calls CAPI to determine which method to run, while also passing in values for necessary parameters.
4. CAPI interfaces with the API of the cloud provider to provision the actual resource.
Process flow when a resource is created in the cloud provider

1. The cloud provider provisions the resource and responds with JSON.
2. The system parses the response inside the resource script in the resource block.
3. The system updates the CMDB. When provisioning a new resource, the system creates a CI in the correct table.
4. The new stack appears in the cloud user portal for the user to manage.

The Resource Blocks page

Manage resource blocks in the Resource Blocks page. Open the Cloud Admin portal, and then navigate to **Design > Resource Blocks**.
Create a custom resource block

If the blocks in the base system do not provide the cloud resources that you need to provision, you can create a custom resource block.

Role required: sn_cmp.cloud_service_designer

1. Navigate to **Cloud Management > Cloud Admin Portal > Design > Resource Blocks** and click **New**.
2. Enter a unique and meaningful **Name** and **Description** for the resource block, and then enter the following settings:
### Layer

Select the layer that represents the type of resource block:

- **Application**: Resources that are applications. For example, the 'PetStore' application might exist on a Tomcat or WebLogic platforms.
- **Platform**: Resources that are typically hosted on a virtual server like MySQL, Tomcat, or Apache.
- **Logical DataCenter**: Resources that are datacenters like AWS, Azure, or vSphere.
- **Virtual Infrastructure**: Resources that are virtual and still provide infrastructure for the virtual world like NetApp SVM, NetApp CDOT, CISCO ICF Cloud, or VMware NSX.
- **Virtual Workload**: Resources that are hosted on a datacenter. For example, Virtual Server, Storage Volume, Network, or Gateway.

**Note:**

- Applications are deployed on Platforms.
- Platforms are deployed on Logical Datacenters or Virtual Infrastructures.

### CI type

Select the type of CI in your CMDB that the resource block represents. When the user launches a stack, the cloud resources are provisioned and actual CIs that for the resources are created in the CMDB in this CI type. For example, if you select Storage Volume, the system creates a record in the Storage Volume (cmdb_ci_storage_volume) table when the user provisions a new volume.
| Default Guest Interface | Create interface | Select the guest interface that contains the *operations* you need for the resource. Later you can further define and select the operations to use for this default guest interface. The system gives you a set of default operations for each interface.

If another resource block specifies a *Host interface* (see the Host interface field description) that matches what you specify here in the *Guest interface* field, that resource block can be a potential host for this resource block.

If you select the *Create interface* check box, enter a name for the guest interface. The system creates the guest interface, opens the resource block that you are configuring in the resource block form, and lists the new interface in the *Resource Guest Interface* related list. See the description of the Resource Guest Interface in the following steps.

If you create a new guest interface, also keep in mind that other resource blocks can use the interface.

| Host interface | Select the interface that the potential host of this resource block must offer up. Any other resource block that offers up a default guest interface of the same type you specify here can become a host of this resource. Later on, you can select one or more actual host resources.

For example, by default the AWS Datacenter resource block specifies the *Compute* interface as a default guest interface. That means the AWS Datacenter is offering the *Compute* host interface to other resource blocks through which to connect to the AWS Datacenter.

Specify the *Compute* interface here as the host interface of the resource block so the resource block can look for any other resource block using the *Compute* interface as its guest interface. Since the AWS Datacenter does use the *Compute* interface as its guest interface, later you can select the AWS Datacenter as the actual host in the Host Resource related list.

Other resource blocks that offer the *Compute* interface by specifying *Compute* as the guest interface, are also potential hosts for this resource block.

You must know which interfaces the potential host resources offer as guest interfaces. And determine which operations are part of those guest interfaces. |
3. Click Submit. Continue updating settings on the related lists, as needed.

4. Optional: Complete the following steps to add another guest interface. You can specify an additional guest interface to offer to other resource blocks. For example, a datacenter resource block might offer a Compute interface, a Network Service Container interface, and a Firewall interface for other resource blocks to use.
   a) Under Guest Interface, click New.
   b) In the Guest interface field, select an interface.
   c) Click Submit or Update.

5. In Host Resource, specify the resource that can host this resource block, typically the one that offers the interface you specified in the Host Interface field.
   The selected host resource block handles the provisioning of this resource block. For example, the host resource is a virtual machine resource block that can host an Apache Web server. The host resource selection limits the options that are presented to users when they select a host.
   For example select vSphere Datacenter in the Host Resource field for this resource to be hosted in a VMWare cloud.
6. Optional: Complete the following steps to add bindings.

A binding is another resource block that this resource block can connect to. Select resource blocks that support the same **Guest interface** configured for this resource block.

When you create a blueprint, you can add an attach connection between two resource blocks that are connected with this binding. Understanding the direction of the binding is essential to configuring the resource block. You attach a virtual server to a network, or you attach storage to a virtual server. You cannot reverse these directions.

a) Under **Bindings**, click **New**.

b) In the **Host Resource** field, select a resource in the **Guest Resource** field.

   The guest resource is the resource block that you are attaching to the current resource block.

c) Click **Submit** or **Update**.

For example, a virtual server can connect to virtual storage. By default, the base system provides a Virtual Server resource block with a binding to the Storage Volume resource block to enable this connection.

---

**Note:** You can select only the host resources that support the **Host interface** previously specified for this resource block.
7. Complete the following steps to configure endpoint operations between the two resources to bind.

The endpoints call operations during binding. For example, a storage volume can specify an endpoint with required conditions and the operations to execute to make a connection to a virtual server. The virtual server must also support a binding interface to the storage volume.

Implement an endpoint type, such as **Block EP**, for the virtual server to consume the endpoint. Later, when you create blueprints, create the binding between the storage volume and the virtual server.

![Diagram of storage volume and virtual server binding](image)

**Example of binding in a blueprint**

a) Under **Endpoint Operations**, click **New**.

b) In **Binding Resource**, select the resource block that consumes the endpoint from this resource block.

c) Select the type of endpoint that the **Source Resource** presents.

d) In **Operation implementation**, select the implementation that performs the operation you are defining in this endpoint mapping.

e) Select the **Type** of operation to perform for the mapping.
   Only operations that are appropriate for the specified **Endpoint** appear in the list.
   Do not change the settings in **Source Resource**.

f) Click **Submit** or **Update**.

Specify a host resource for a resource block
Hosts that support the Host interface of a resource block are potential hosts for the resource block. You use the Host interface setting to further limit the options that are presented to the stack requester while selecting a host type.

Role required: admin

1. In the Cloud Admin Portal, navigate to Design > Resource Blocks.
2. Select the resource block and then select Draft on the Resource Blocks form:

3. In the Host Resource related list, click New.
4. On the Resource Host form, the Application and the Guest Resource (the resource block that you are currently configuring) are already specified. Specify the Host Resource. For example, for a virtual server, you might select AWS Datacenter or Azure Datacenter as potential host types.
5. Click Submit.

Specify bindings for resource blocks

Bindings represent endpoint relationships. For example, a storage volume might implement an endpoint type of Block EP (cmdb_ci_endpoint_block). A virtual server might consume an endpoint of that type. Bindings must support the Guest interface that is specified for the resource block.

Role required: admin

1. In the Cloud Admin Portal, navigate to Design > Resource Blocks.
2. Select the resource block and then select Draft on the Resource Blocks form:

3. In the Resource Bindings related list, click New.
4. On the Resource Binding form, the Application and the Guest Resource (the resource block that you are currently configuring) are already specified. Specify the Binding Resource.
5. Click Submit.

Configure endpoint operation mapping

You can configure endpoint mapping on a storage volume to make a connection to a resource.

Role required: admin

In the following example, the endpoint mapping that is configured for a storage volume specifies the conditions that must be met and the operation to execute to make a connection to a resource (a virtual server in the example) that supports a binding interface to a storage volume. The storage volume implements the endpoint (type Block EP in the example) and the virtual server consumes the endpoint:
The settings on the example **Resource Endpoint Mappings** related list specifies the conditions and an action:

- For a resource block that supports a binding interface to a **Storage Volume**
- For a **Source** resource with **CI type** of **Storage Volume**
- With an **Endpoint** type of **Block EP** (cmdb_ci_endpoint_block)

Use the **Attach** operation to implement the connection. Use the specified **Operation Implementation** to perform the **Attach** operation.

1. While configuring a resource block, open the **Resource Endpoint Mapping** related list, click **New**, and then specify the following settings:

   | Binding Resource | Resource block that consumes the **Endpoint** that the **Source Resource** presents. |
2. Repeat the process for as many mappings as are needed.

Configure resource block input parameters
You can configure operations for each interface of a resource block by specifying input parameters, steps, and output attributes. Input parameters hold values that the system requires to identify and manage a virtual resource, such as the datacenter and resource group that the resource is in.

Role required: sn_cmp.cloud_service_designer

The system automatically associates default operations from the Cloud API (CAPI) to the resource block when you add interfaces to it. It also provides default input parameters for each operation.

1. Open a resource block, and then click the **Operations** tab.

   ![Operations tab](image)

   The **Input Parameters** sub-tab appears by default.

2. Select the interface from the **Interface** choice list.

   ![Interface choice list](image)

   The interfaces in this choice list are derived from the **Resource Guest Interface** list on the **General Information** tab that you already configured when you created a resource block.

3. Select the operation on which you want to modify the input parameters. Optionally, you can create a new operation:
   a) Click the **Add Operation** icon.
b) Fill out the form (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a descriptive name.</td>
</tr>
<tr>
<td>Operation type</td>
<td>Select the type of operation from the choice list.</td>
</tr>
<tr>
<td>Access type</td>
<td>Select <strong>Public</strong> to make the resource block operation available from the catalog order form when a user manages a stack that they own. The actual operations appear in a list on the catalog order form: Select <strong>Private</strong> to prevent the operation from being used when a user launches a stack. This setting applies at the catalog-item level, not the whole-stack level. You can control the access type on the whole stack when you configure blueprints. As an example, you might want to mark the Deprovision operation as private for an individual VM, but not for the whole stack. This allows the user to deprovision everything at once and prevents the user from just deprovisioning the VM.</td>
</tr>
</tbody>
</table>

c) Click **Submit**.
4. Configure the input parameters to provide a way to input the items you need to run the selected operation.

For example, if you add a Provision operation for an AWS resource, you need, at minimum, the CloudAccount parameter and the Location parameter to perform the operation. You need to know the providers API to know what to pass to it. For example, if you want to add a provision operation for an S3 bucket on AWS, you need to know the allowable AWS operations on buckets and the Request Elements for things like creating a bucket (the PUT Bucket operation). You can typically find these resources in the AWS and Azure documentation.

a) Click the **Add Input Parameters** icon.

![Add Input Parameters icon](image)

b) Add the **Name** of the parameter that you obtained from the cloud provider API. In this example, you could add CreateBucketConfiguration for an AWS S3 bucket.

c) In the **Datasource** column, select the type of input parameter you are configuring. This value specifies the source or location where the parameter value comes from:

- **Text**: a text value that the user passes.
- **Pools**: the query that you want to run against a pool to get a value back from it. The pool **Datasource Value** should follow this convention: `ServiceNow::Pools::PoolName.Pool-Filter-Name`. See [Pool governance for Cloud Management](#) for more information on pools.
- **CSVList**: values from a comma-separated value list that you enter into the **Datasource Value**.
- **Array**: an array of values. You can programmatically send it an array in JavaScript. The format is as follows: `['abc','def']`.
- **FormData**: data submitted from the form configured in the blueprint.
- **ResourceScript**: an expression that passes a value. The expression should follow this convention in the **Mapping** field:

```javascript
$(Script:script-name.function[arg1=${value-to-pass},arg2=${value-to-pass}])
```
- **TLR**: the value inherited from the top-level resource container above this resource block in the blueprint.

d) Click **Save**.

e) In the **Mapping** column, you can view the auto-generated mapping values. Mappings specify where to pull the information in the system. You do not have to change the mapping values except for **ResourceScript** type parameters, where you must enter the expression.

f) In the **Default Value** column, specify a value that you want the system to use if the user does not pass any value or if you did not enter anything in the **Datasource Value**.

g) Select **Create Form** to make this attribute available in the order form that the user sees when trying to provision the stack item. You can also select **Mandatory** to force the user to enter a value.
In the Restricted To column, click the restricted icon (🔒), and then select a resource block to restrict the attribute to be passed to a specific host.

In this example, you are configuring a resource block for a virtual server on vSphere, which requires the name of the VM folder (represented by the `VmFolderName` attribute). Other datacenter types do not need the name of the VM folder. Therefore, you can restrict `VmFolderName` to the vSphere datacenter resource block.

This image illustrates how the `VmFolderName` attribute is restricted to the vSphere Datacenter.
Add operation steps to a resource block
After you configure operations and input parameters for your resource block, add steps for each operation to tell the system which Cloud API (CAPI) to call. Each step is a separate call to CAPI.

Role required: sn_cmp.cloud_service_designer

1. On the resource block, click the **Steps** sub-tab under **Operations**.
2. Click **Add Step**.
3. Fill in the form fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td>Select the type of operation:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Invoke Cloud API</strong>: Select this option to select a provider, interface, and method from CAPI.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Invoke Host Resource Operation</strong>: Select this option to use an operation (also called an operation signature) from the host interface that you already specified for this resource block.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Invoke Script</strong>: Select this option to invoke a method from an existing MID Server script include.</td>
</tr>
</tbody>
</table>

<p>| API Provider (Operation Type = Invoke Cloud API) | Select the API that provides the interfaces you need. The providers are part of the Cloud API (CAPI). |
|                                                 | For example, CAPI supplies the AWS provider, which has products, some of which have provider names. |
|                                                 | This value appears if you select the Invoke Cloud API operation type. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API Interface</td>
<td>Select the interface that has the methods you want to use. For example, the Blob Storage Interface provides the CreateContainer interface, which the system uses during provisioning.</td>
</tr>
<tr>
<td>API Method</td>
<td>Select the method to perform the operation. In this example, CreateContainer is required to provision storage.</td>
</tr>
<tr>
<td>CAPI Version</td>
<td>Enter the version of the CAPI API.</td>
</tr>
<tr>
<td>Host Interface</td>
<td>Select the interface from which to call an operation. You can only select the same interface that you configured on the General Information tab of the resource block. To change the interface, navigate back to the General Information tab and select another value in the Host Interface field.</td>
</tr>
<tr>
<td>Interface Operation</td>
<td>Select the operation from the host interface. For example, if you select Storage Server Interface, you can call all the operations in that interface, such as CreateStorageVolume.</td>
</tr>
<tr>
<td>Scripts</td>
<td>Enter the script you want to invoke, along with a method in the script. Use this syntax: script-name.method.</td>
</tr>
<tr>
<td>Workflow</td>
<td>Do not use Workflow.</td>
</tr>
<tr>
<td>Condition</td>
<td>Enter an optional condition that must be met before the system uses this step.</td>
</tr>
</tbody>
</table>

If you look at the Azure API and the CreateNode method mapper, you can see that the endpoint operation is a script. It calls the MID Server script include, which shows you the createNode() function. This function uses AzureComputeVirtualMachine script include. Take time to review and understand this script include.

4. Click Submit.

The Input and Response Processor tabs appear for the step.

5. On the Input tab, configure the CAPI operation inputs.

On the CAPI interface, which in this case is Blob Storage Interface, open CreateContainer method. The inputs from CAPI are shown on the Input tab, where you create parameters.

Remember that the parameters on the Input Parameters tab is where users enter values. You can map those input parameters to the parameters on the Input tab. To do this, copy the Mapping value from the Input Parameters tab into the Value field on the Steps > Input tab.

If this step is invoking a script, and that script requires arguments to return values, you can add a new input parameter for each argument. Use this syntax in the Value field: $(parameter.arg1).

6. If any subsequent steps in this operation or subsequent operations require parameters values from this step to be passed to them, configure output attributes.
For example, the Provision operation for a virtual server can provide an output attribute for the network interface ID and another for the node ID. When the cloud provider sends the response, the actual ID of the network and node are parsed out of the response and saved in the output attributes. Subsequent steps use the IDs to perform actions.

a) Click the **Output Attributes** tab.

b) Enter code in the following notation:

```json
{
   "Attribute": "${Value}"
}
```

Where the value comes from a mapping to the interface and the operation.

For example, in the following output attribute:

```json
{
   "NodeId": "${Compute Interface.CreateNode.Output.resp.nodeId}"
}
```

- **NodeId** is the attribute.
- **Compute Interface** specifies the interface.
- **CreateNode** is the name of the method that creates the node.
- **Output.resp** is a mandatory part of the syntax.
- **nodeId**

Add an operation to a resource block to use with a workflow

You can select from the existing operations in a resource block to work with your workflow or you can create a custom operation.

**Role required:** Cloud designer or admin

1. In the Cloud Admin Portal, navigate to **Design > Resource Blocks**.
2. Select a resource block, for example **Virtual Server**.
3. Move the state of the resource block from **Published** to **Draft** to make it editable.
4. Select the **Operations** tab.
5. From the **Interface** drop-down field, select an interface that is available to the user. The default interface is **Virtual Server Interface**.
6. Click + to the right of the **Operation** field, and perform these steps in the Add Operation dialog:
   a) Enter a meaningful and unique name.
   b) Select an operation from the **Operation Type** drop-down.
   c) Select **Public** from the **Access Type** field.
7. Fill in the form, and then click **Submit**.

<table>
<thead>
<tr>
<th>Operation Type</th>
<th>Select the operation to perform.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Type</td>
<td>For the user to see the operation at the Resources level during provisioning, select <strong>Public</strong>. Operations set to <strong>Private</strong> are not visible during provisioning.</td>
</tr>
</tbody>
</table>
The operation is available on the resource when you select the resource for your workflow.

**Add a resource operation step to invoke a workflow**

Add a resource operation step to invoke a workflow

You can invoke a workflow by adding an operations step to a resource and then associating the resource to a workflow.

- Role required: Cloud user, designer, or admin
- The Orchestration application is installed.

1. In the Cloud Admin Portal, navigate to **Design > Resource Blocks**.
2. Click the resource block to add an operations step to.
3. Set the resource block to **Draft** state and then click **Operations > Steps > Add Step**.
   
   To enable the workflow to be exported as part of the update sets, you must create an extension interface and add the new operation. See [Extend Cloud Management resource blocks with an override operation](...) for details.

4. On the Add Operation Steps popup, select **Invoke Workflow** for **Operation Type**.
5. Select the **Workflow** from the list of workflows that you created.

The system can filter the list using tags. To add a tag:

1. Navigate to **System Properties**.
2. Locate and select the system property **sn_cmp.workflow_tag_filter**.
3. Edit the property. Add a tag or comma-separated list of tags.
4. Click **Save**.

6. To filter the workflows:
   a) On the workflow table, open the workflow.
   b) Click **Add Tag**.
   c) Enter the value that you provided earlier in the property **sn_cmp.workflow_tag_filter**
   d) Click **Enter**.
   
   The value is added as a tag to the workflow.

The new step appears after the software generates the new Day 2 Operation Catalog.

7. Workflows can exist that are created on tables other than global. You can add a workflow created on such a table.

To execute operations on this type of workflow, you need the sys_id of the record in which the workflow is executing to create the current record. So when the designer adds a workflow that is on table other than global, the system creates a **wf_current** parameter in the operation input parameter. You can then write an expression to map the sys_id of the resource in which the operation is executing. Once this is completed, the designer can use the ‘current’ key word in workflow scripts.

To add the workflow:

a) After you create the step, the system adds the workflow inputs to the operation input parameters.

b) You can then provide the mapping for these parameters, if needed.

c) Click the **Generate Catalog** button to create the catalog item for the operation.

d) After the catalog generates, you can add from the load and field change rules to the catalog items.
8. Set the resource block to the **Published** state to make the workflow available in the Cloud User Portal.

Extend Cloud Management resource blocks with an override operation
You can extend your existing guest interface to create a new operation. You can also create a new guest interface with its own operations, parameters, and inputs.

- Role required: sn_cmp.cloud_admin
- Become familiar with resource block input parameters, operations, and steps. See [Configure resource block input parameters](#) and [Add operation steps to a resource block](#) for instructions.

The Cloud Management application is bundled with several useful resource blocks. Create extensions to the existing resource block interfaces to add more functionality to these resource blocks. After creating an interface extension, use it in a blueprint to execute a customized operation. For example, if you do not like the way a virtual server is provisioned, extend the virtual server resource block interface by adding operations or overriding existing operations.

![Note: Operations in resource blocks are also called operation signatures.](#)

1. Open a resource block.
2. Change the status of the resource block from **Published** to **Draft**.
3. Click the **Operations** tab.
4. Click the **Override Operation** icon.

5. Select an existing guest interface or select the **Create new** check box to create a new guest interface.
6. Optional: If you create an interface, enter a name in the **Interface** field.
7. Click **Submit** or **Update**.
   The new operation appears in the **Operation** list. If you create a new interface, it appears in the **Interface** list.
8. Configure the necessary input parameters, steps, and output attributes for your new guest interface.
9. To edit the operation or to change the access type:
   a) Click the **Edit Operation Signature** icon.
b) Change the **Operation Type** if necessary, and add, edit, or remove input parameters at the bottom.

c) If you want the blueprint designer to perform the operation directly on the resource only when it is instantiated, select **Private** for the **Access Type**.

d) Click **Update**.

10. After completing the customizations, toggle the switch to **Published**.

11. Navigate to **Cloud Management > Cloud Admin Portal > Design > Blueprints**.

12. Follow the steps in [Build a Cloud Management blueprint](#) to add a blueprint.

13. Click **Operations** and click the operation to modify in the resource block. The default resource block interface appears in the right column.

14. Click the **Edit Resource Operation** (pencil) icon to apply the interface extension you created.

15. Click **Submit**. The blueprint gets executed with the customized interface extension.

After you add this resource block to a blueprint, the parameters you added are available for you to customize.

**Add a resource script to a resource block**

A resource script operates on a resource during deployment or returns data to the CMDB after a resource is deployed.

Role required: sn_cmp.cloud_service_designer

1. In the Cloud Admin Portal, navigate to **Design > Resource Blocks**.

2. Open the resource block and then set it to **Draft** state.

3. On the **Resource Script** tab, specify a unique and meaningful **Name** and then define the script:

   ```
   Script code.
   ```
Configure a response processor

Configure a response processor to handle responses from datacenters.

- The CI class types for your resources must already exist in the CMDB. If you created custom products or providers in CAPI, you must also create custom CI types so the response processor knows how to identify CIs. See Create a CI class for a virtual cloud resource for information on creating CI types.
- Role required: sn_cmp.cloud_service_designer

After each step is completed, the datacenter sends a response to the ECC Queue of the MID Server. For example, if a user provisions a virtual server on an AWS datacenter, AWS sends information about the new virtual server to the ECC Queue. The system needs to create a CI in the CMDB for the virtual server. It does this using a response processor.

1. Open a resource block, and click the Resource Script tab.
2. Click New.
3. Enter a descriptive name and select Response Processor in the Type field.
4. In the Script field, enter a script.
5. Click Submit.
6. Click the Operations tab on the resource block.
7. Click the Steps sub tab.
8. Click a step.
9. Click the Response Processor tab for the step.
10. Click the add icon (➕).
11. In the Script Name field, select the processor you just created.
12. Click Submit.

The response processor script can process the response from the cloud provider for the CI type of the resource block.

Execute response processor for workflow

Execute a response processor for a workflow to get the workflow data back into a configuration item (CI). The response processor picks up the data, sends the data to the CMDB which in turn puts the data in a CI.
Role required: sn_cmp.cloud_service_designer.

Before you execute a response processor for a workflow, you need to create a workflow, attach the workflow to a resource block operation step, and then generate the catalog. To return the response from the workflow to the Cloud Management application, the workflow designer needs to add the Cloud Return Response activity to the workflow. A variable needs to be mentioned inside the Cloud Return Response activity for the response. See Workflow.

1. In the Cloud Admin Portal, navigate to Design > Resource Blocks.
2. Open a resource block that is in a draft state and navigate to OperationsSteps.
3. Click Add Step.
   The Add Operation Steps dialog box appears.
4. Add a workflow operation step. See Add operation steps to a resource block.
   The workflow operation step gets attached to the resource block and appears on the page. Any input parameters associated with the workflow appear on the Input tab.
5. Click Generate Catalog.
   The workflow appears in the Cloud User Portal as an operation. Select the workflow from the Select Operation picker to execute the operation. The status of the operation is visible in the Track operation sub tab.
6. Click the Response Processor tab and then click the plus icon.
   The Add Response Processor dialog box appears.

7. In the Script Name list, select a script for the response processor.
   For a script to appear in the Script Name list, the script should already have been created in the Resource Script tab.
8. Click Submit.
   The script appears in the Response Processor tab. You can open the script and make modifications to the script.

Virtual server response processor example
The Create_Virtual_Server_Response_Processor script, which is available by default in Cloud Management, is the response processor that handles the creation of AWS virtual machine CIs.

Create Virtual Server

The Create_Virtual_Server_Response_Processor resource processor script is available by default on the Virtual Server resource block. Its job is to create a virtual server record in the Virtual Machine Instance (cmdb_ci_vm_instance) table when a new virtual server is provisioned.
All response processors have this first line with these common parameters:

```javascript
function processResponse(response, cloudServiceAccountId, ldc, correlationId, step, requestorContext) {
    // This brings in the response from the cloud provider and the important information, such as the account ID, that is required for the new CI that the system can create. All of these parameters are required for all response processors.

    // Line 10 parses the response into JSON so that the system can process it. The information is held in the vmResponse variable:

    var vmResponse = global.JSON.parse(response);

    // Whenever you create or edit a response processor, you must know which inputs are required for the CI type. Line 11 handles one of the necessary inputs, the hardware ID, that the CMDB record requires:

    var hardwareId = vmResponse.hardwareId;

    // Line 39 shows the information that is required for the system to identify the new virtual server and related CIs, so the information can be put into the CMDB. In this case, the service account object ID identifies the account associated with the virtual server, the datacenter object ID identifies the datacenter in which the virtual server lives, and the virtual machine instance object ID identifies the virtual server itself. This identification code block prevents the creation of duplicate CIs.

    var vmInfo = {
        "cmdb_ci_vm_instance": {
            "validator": "virtual_machine_create_update_validator",
            "validator_overrides": {},
            "identification": {
                "cmdb_ci_cloud_service_account": {
                    "criterion": {
                        "object_id": cloudServiceAccountId
                    }
                },
                "cmdb_ci_aws_datacenter": {
                    "criterion": {
                        "object_id": ldc
                    }
                },
                "cmdb_ci_vm_instance": {
                    "criterion": {
                        "object_id": vmResponse.nodeId
                    }
                }
            }
        }
    }

    // Attributes are populated into the fields on in the cmdb_ci_vm_instance table. These attributes are defined in line 61:

    "attributes": {
        "name": vmResponse.nodeName,
        "object_id": vmResponse.nodeId,
        "state": status_map[vmResponse.state],
        "dns_suffix": vmResponse.dnsSuffix,
        "cpus": vmCPUs,
    }
```
References to other CIs can also be included in the response processor. In this case, the OS template that the virtual server is based on is identified by first identifying the object ID of the service account and the datacenter along with the actual OS template.

```
"references": {
  "cmdb_ci_os_template": {
    "identification": {
      "cmdb_ci_cloud_service_account": {
        "criterion": {
          "object_id": cloudServiceAccountId
        }
      },
      "cmdb_ci_aws_datacenter": {
        "criterion": {
          "object_id": ldc
        }
      },
      "cmdb_ci_os_template": {
        "criterion": {
          "object_id": imageIdTrim
        }
      }
    }
  }
},

The following code block adds the object ID of the OS image to the attributes list so that this information can be populated into the virtual server CMDB record.

```
"attributes": {
  "object_id": imageIdTrim
}
```

This code block performs additional identification on the Compute template (the hardware type) and then add it to the attributes:

```
"cmdb_ci_compute_template": {
  "identification": {
    "cmdb_ci_cloud_service_account": {
      "criterion": {
        "object_id": cloudServiceAccountId
      }
    },
    "cmdb_ci_aws_datacenter": {
      "criterion": {
        "object_id": ldc
      }
    },
    "cmdb_ci_compute_template": {
      "criterion": {
```
Additional code sections make the relationship with network interfaces and identify any storage attached to the virtual server.

This mandatory code block pushes the data to the CMDB and returns the JSON string:

```
cloudModelString.push(vmInfo);
return global.JSON.stringify(cloudModelString);
```
### Field

<table>
<thead>
<tr>
<th>Script</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Script for Create</td>
<td>Enter the following script:</td>
</tr>
<tr>
<td></td>
<td>function processResponse</td>
</tr>
<tr>
<td></td>
<td>(response,</td>
</tr>
<tr>
<td></td>
<td>cloudServiceAccountId, ldc,</td>
</tr>
<tr>
<td></td>
<td>correlationId, step,</td>
</tr>
<tr>
<td></td>
<td>requestorContext, stackId) {</td>
</tr>
<tr>
<td></td>
<td>var responseObject =</td>
</tr>
<tr>
<td></td>
<td>global.JSON.parse(response);</td>
</tr>
<tr>
<td></td>
<td>var tfState =</td>
</tr>
<tr>
<td></td>
<td>global.JSON.parse(responseObject.terraform.state);</td>
</tr>
<tr>
<td></td>
<td>var processor = new</td>
</tr>
<tr>
<td></td>
<td>sn_cmp_terraform.TerraformResponseProcessor();</td>
</tr>
<tr>
<td></td>
<td>var processedResponses =</td>
</tr>
<tr>
<td></td>
<td>processor.processResponse(tfState,</td>
</tr>
<tr>
<td></td>
<td>cloudServiceAccountId, ldc,</td>
</tr>
<tr>
<td></td>
<td>correlationId, step,</td>
</tr>
<tr>
<td></td>
<td>requestorContext, stackId);</td>
</tr>
<tr>
<td></td>
<td>return</td>
</tr>
<tr>
<td></td>
<td>global.JSON.stringify(processedResponses);</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td>Script</td>
<td>Enter the following script:</td>
</tr>
<tr>
<td>Resource Script for Delete</td>
<td>function processResponse</td>
</tr>
<tr>
<td></td>
<td>(response,</td>
</tr>
<tr>
<td></td>
<td>cloudServiceAccountId, ldc,</td>
</tr>
<tr>
<td></td>
<td>correlationId, step,</td>
</tr>
<tr>
<td></td>
<td>requestorContext, stackId) {</td>
</tr>
<tr>
<td></td>
<td>var responseObject =</td>
</tr>
<tr>
<td></td>
<td>global.JSON.parse(response);</td>
</tr>
<tr>
<td></td>
<td>var tfState =</td>
</tr>
<tr>
<td></td>
<td>global.JSON.parse(responseObject.terraform.state);</td>
</tr>
<tr>
<td></td>
<td>var processor = new</td>
</tr>
<tr>
<td></td>
<td>sn_cmp_terraform.TerraformResponseProcessor();</td>
</tr>
<tr>
<td></td>
<td>var processedResponses =</td>
</tr>
<tr>
<td></td>
<td>processor.processResponse(tfState,</td>
</tr>
<tr>
<td></td>
<td>cloudServiceAccountId, ldc,</td>
</tr>
<tr>
<td></td>
<td>correlationId, step,</td>
</tr>
<tr>
<td></td>
<td>requestorContext, stackId);</td>
</tr>
<tr>
<td></td>
<td>return</td>
</tr>
<tr>
<td></td>
<td>global.JSON.stringify(processedResponses);</td>
</tr>
</tbody>
</table>

5. Click **Submit**.
6. Click the **Operations** sub tab and then click **Steps**.
7. Select **CreateStack** from the **Operations** list.
8. Click the **Add a Step** plus icon to open the Add Operation Steps dialog box and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td>Select the type of operation:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Invoke Cloud API</strong>: Select this option to select a provider, interface, and method from CAPI.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>API Provider</strong></td>
<td>Select the API that provides the interfaces you need. The providers are part of the Cloud API (CAPI). For example, CAPI supplies the Terraform OpenSource provider, which provides the Configuration Orchestration Interface. This value appears if you select the Invoke Cloud API operation type.</td>
</tr>
<tr>
<td><strong>API Interface</strong></td>
<td>Select the interface that has the methods you want to use. For this example, the Configuration Orchestration Interface provides multiple interfaces or methods, which the system uses during provisioning.</td>
</tr>
<tr>
<td><strong>API Method</strong></td>
<td>Select the method to perform the operation. In this example, <code>ExecuteCommand</code> is required to provision terraform templates.</td>
</tr>
<tr>
<td><strong>CAPI Version</strong></td>
<td>Enter the version of the CAPI API.</td>
</tr>
<tr>
<td><strong>Condition</strong></td>
<td>Enter an optional condition that must be met before the system uses this step.</td>
</tr>
</tbody>
</table>

9. Click **Submit**.
10. Click **Response Processor** and then click the plus icon. The Add Response Processor dialog box appears.
11. Select **Create_AzureTF_Stack_Response_Processor** from the **Script Name** list and click **Submit**.
12. Move the state of the resource block from **Draft** to **Published**. The Azure Datacenter resource block moves from the draft to the published state and no further changes can be made to the resource block. Changes to a resource block can be made only if the status is set to **Draft**.

### Add Terraform Orchestration interface on a resource block

After you have activated the Terraform store app for Cloud Management, to support cloud-based operations for cloud providers using Terraform Open Source Edition. Add the Terraform Orchestration interface on resource block, to execute cloud-based operations via terraform templates.

Role required: `sn_cmp.cmp_root_admin`, `sn_cmp.cloud_service_designer`

1. Navigate to On the Cloud Admin Portal, navigate to **Design > Resource Block**.
2. Click the resource block, for example **Azure Datacenter**.
3. Move the state of the resource block from **Published** to **Draft** to make it editable.
4. Navigate to **General Information > Guest Interface**, and then click **New**.
5. Search for and select the **Terraform Orchestration Interface** from the Guest Interface list, and click **Submit**.
6. Click the **Operations** tab.
7. In the **Input Parameters** tab, navigate to **Interface** and select **Terraform Orchestration Interface**
8. On the **Input Parameters** tab, configure the CAPI operation inputs.
On the CAPI interface, which in this case is Terraform Orchestration Interface, open CreateStack method. The inputs from CAPI are shown on the Input tab, where you create parameters.

Remember that the parameters on the Input Parameters tab is where users enter values. You can map those input parameters to the parameters on the Input tab. To do this, copy the Mapping value from the Input Parameters tab into the Value field on the Steps > Input tab.

If this step is invoking a script, and that script requires arguments to return values, you can add a new input parameter for each argument. Use this syntax in the Value field: ${parameter.arg1}.

9. Click the Steps sub-tab under Operations.
10. Click Add Step.
11. Fill in the form fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td>Select the type of operation:</td>
</tr>
<tr>
<td></td>
<td>• Invoke Cloud API: Select this option to select a provider, interface, and method from CAPI.</td>
</tr>
<tr>
<td>API Provider</td>
<td>Select the API that provides the interfaces you need. The providers are part of the Cloud API (CAPI). For example, CAPI supplies the Terraform OpenSource provider, which provides the Configuration Orchestration Interface. This value appears if you select the Invoke Cloud API operation type.</td>
</tr>
<tr>
<td>API Interface</td>
<td>Select the interface that has the methods you want to use. For this example, the Configuration Orchestration Interface provides multiple interfaces or methods, which the system uses during provisioning.</td>
</tr>
<tr>
<td>API Method</td>
<td>Select the method to perform the operation. In this example, ExecuteCommand is required to provision terraform templates.</td>
</tr>
<tr>
<td>CAPI Version</td>
<td>Enter the version of the CAPI API.</td>
</tr>
<tr>
<td>Condition</td>
<td>Enter an optional condition that must be met before the system uses this step.</td>
</tr>
</tbody>
</table>

12. Click Submit.

The Input and Response Processor tabs appear for the step.

13. On the Input tab, configure the CAPI operation inputs.

On the CAPI interface, which in this case is Configuration Orchestration Interface, open ExecuteCommand method. The inputs from CAPI are shown on the Input tab, where you create parameters.

Remember that the parameters on the Input Parameters tab is where users enter values. You can map those input parameters to the parameters on the Input tab. To do this, copy the Mapping value from the Input Parameters tab into the Value field on the Steps > Input tab.
If this step is invoking a script, and that script requires arguments to return values, you can add a new input parameter for each argument. Use this syntax in the Value field: `${parameter.arg1}`.

**Note:** Please map the values as mentioned below for the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdditionalParameters</td>
<td>Enter the following script:</td>
</tr>
<tr>
<td></td>
<td>{&quot;ServerType&quot;:&quot;${Script:sn_cmp_terraform.TerraformCommandUtils.</td>
</tr>
<tr>
<td></td>
<td>getConfigProviderServerTypeByName[arg=${parameter.WorkloadConfigProvider}]&quot;}</td>
</tr>
<tr>
<td>ConfigMgmtProviderInfo</td>
<td>$(capiResolver.NodeCredentialResolver#nodeCredential[$ci.sn_cmp_wl_cfg_mgmt_provider[name=${parameter.WorkloadConfigProvider}].credential])</td>
</tr>
<tr>
<td>Node Address</td>
<td>$(ci.cmdb_ci_workload_config_provider[name=${parameter.WorkloadConfigProvider}].url)</td>
</tr>
<tr>
<td>Script</td>
<td>For CreateStack operation:</td>
</tr>
<tr>
<td></td>
<td>${CloudScript.scripts.TerraformDeploy}</td>
</tr>
<tr>
<td></td>
<td>For DeleteStack operation:</td>
</tr>
<tr>
<td></td>
<td>${CloudScript.scripts.DestroyTerraform}</td>
</tr>
</tbody>
</table>

14. On the **Response Processor** tab, configure the response processors by attaching resource scripts. You would have added Resource Scripts to the resource block, while creating the Resource Block.

15. From the Operation list, select the **CreateStack** or **DeleteStack** operation.

16. Click the add icon for **Add Response Processor**.

17. Click the **Resource Script** sub tab, and then click **New**.

18. Complete the form.

**New Record**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (For example, <strong>Create_AzureTF_Stack_Response_Processor</strong>)</td>
</tr>
<tr>
<td>Type</td>
<td>Select a type (<strong>Response Processor</strong>) for the script.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Script</td>
<td>Enter the following script:</td>
</tr>
<tr>
<td>Resource Script for Create</td>
<td>function processResponse (response, cloudServiceAccountId, ldc, correlationId, step, requestorContext, stackId) { var responseObject = global.JSON.parse(response); var tfState = global.JSON.parse(responseObject.terraform.state); var processor = new sn_cmp_terraform.TerraformResponseProcessor(); var processedResponses = processor.processResponse(tfState, cloudServiceAccountId, ldc, correlationId, step, requestorContext, stackId); return global.JSON.stringify(processedResponses); }</td>
</tr>
<tr>
<td>Script</td>
<td>Enter the following script:</td>
</tr>
<tr>
<td>Resource Script for Delete</td>
<td>function processResponse (response, cloudServiceAccountId, ldc, correlationId, step, requestorContext, stackId) { var responseObject = global.JSON.parse(response); var tfState = global.JSON.parse(responseObject.terraform.state); var processor = new sn_cmp_terraform.TerraformResponseProcessor(); var processedResponses = processor.processResponse(tfState, cloudServiceAccountId, ldc, correlationId, step, requestorContext, stackId); return global.JSON.stringify(processedResponses); }</td>
</tr>
</tbody>
</table>

19. From the **Script Name** list, select the script corresponding to the selected operation, and click **Submit**.

20. Move the state of the resource block from **Draft** to **Published**.

The datacenter resource block moves from the draft to the published state and no further changes can be made to the resource block. Changes to a resource block can be made only if the status is set to **Draft**.

**Cloud scripts and cloud script templates**

In the Cloud Management application, script execution is divided into cloud scripts and cloud script templates. Use scripts in blueprints, resource blocks, OS profiles, and use policy scripts to set request form attributes. Policy scripts cannot override user data.
Cloud scripts

Cloud scripts are simple java scripts that use platform features. An example of a sample cloud script:

```javascript
function evaluateTemplate() {
    // Template Resolver is a helper function which we will use to fetch a
    // template and replace its parameters,
    // var templateResolver=new TemplateResolver(); this comes inbuild
    // templateAttributes below will passed from the user.
    var listDir=templateResolver.getTemplate('Install',templateAttributes);
    return listDir;
}
```

Each cloud script contains certain parameters. Parameters are the attributes that you want to access. A parameter contains a name, a default value, and an override value. The default value can be a string literal, a resource block parameter or a complex script expression used in resource blocks.

A cloud script, by default, has access to the `templateResolver` object. The `templateResolver` object has, by default, a `getTemplate` method, whose first parameter is the cloud script template name. `templateAttributes` are the cloud script parameters created as part of a cloud script. If needed, customizations can be done in `templateAttributes`. As an example, see the following code snippet if the value of `DevName` parameter needs to be changed from John to Emily.

```javascript
function evaluateTemplate() {
    // Template Resolver is a helper function which we will use to fetch a
    // template and replace its parameters,
    // var templateResolver=new TemplateResolver(); this comes inbuild
    // templateAttributes below will passed from the user.
    templateAttributes['DevName'] = 'Emily';
    var customNodeName;
    var gr = new GlideRecord('some_custom_table_to_query');
    gr.addQuery('some_parameter','some_value');
    gr.query();
    if(gr.next())
        customNodeName=gr.getValue('some_custom_node_name');
    templateAttributes['NodeName'] = customNodeName;
    var listDir=templateResolver.getTemplate('Install',templateAttributes);
    return listDir;
}
```

Cloud script templates

Cloud script templates are actual executables which are passed to target a virtual machine for execution. Cloud templates can be of any type (Shell/PowerShell/Cloudinit) depending on the execution context. You need to create a cloud template first and then associate it with a cloud script.

Cloud scripts as CloudInit

Cloud providers provide a way to execute scripts at boottime for virtual machines. In the Cloud Management application, you can specify a cloudinit script at various levels:
• **Image level:** At the time of creating a cloud script, you can specify an image against which this script should be executed. Once an OS profile which contains this specific image gets selected, the default cloud script against this image is executed. In the example below, an Apache cloud script is associated with a particular image.

![Cloud Script Template](image)

• **OS profile level:** You can run a cloudinit against a specific cloud provider and against a specific AMI/image. You can also be generic and associate a cloudinit to an OS profile, irrespective of the underlying image. In the following example, multiple images are associated with the OS profile. You can map any cloud script to the OS profile in the Cloud Script OS Profile Mappings tab. The mapping of a cloud script with an OS profile takes precedence over the image-level cloud script association.
In this example, the Apache cloud script is mapped to the Centos OS profile. If needed, you can override the script parameters in the OS Profile Mapping Overrides section.

- Resource blocks: While mapping a cloud script to an OS profile, you can choose whether to execute the cloud script for a specific resource block used in a blueprint and against a particular resource alias. See Create an OS profile.

### Cloud scripts as Postinit

You can use cloud scripts as Postinit scripts in a resource block. Once you create a resource block with a virtual machine to be provisioned, you can add an ExecuteScript operation. The ExecuteScript operation takes a script parameter (a pool of cloud scripts) and you can select any script to execute. Use the `ScriptParameters` attribute in the ExecuteScript operation to override any script parameter. You can provide a JSON map of script parameter and its override value. All expressions that are supported by Cloud Management work in the `ScriptParameters` attribute.

Create cloud initialization script templates and a script

Create cloud initialization templates and a script to use during VM provisioning.

Role required: cloud_admin

1. In the Cloud Admin Portal, navigate to Design > Cloud Scripts > Cloud Script Templates.
2. Click New, enter the name and content of the script, and then click Submit.

Create as many script templates as needed.

3. In the Cloud Admin Portal, navigate to Design > Cloud Scripts.
4. Click New.
5. Enter a descriptive name and create script that uses the template name in the following line of code (`scriptName` is the name of the template):
   
   ```javascript
   templateResolver.getTemplate(scriptName);
   ```

   **Note:** For more information, see Cloud scripts and cloud script templates.

6. Click Submit.

To run a script at the time a virtual resource is provisioned, map the script to an OS profile. To run a post-provisioning script, create a blueprint and specify the script in the Execute Script operation. See Example: Post-provision cloud script for an example.

Map a script to an OS profile

To execute scripted actions during VM provisioning, you can map a script to an OS profile. The script runs on VMs that are created based on the image template in the OS profile.

- Role required: sn_cmp.cloud_admin
- You must have a cloud account with datacenters. You must run Discovery on the service accounts to populate the datacenters.

In this procedure, you specify an existing script. See Create cloud initialization script templates and a script.

**Note:** Profile mappings that specify more details run first. For example, a mapping that specifies a blueprint, OS profile, and resource alias takes precedence over a mapping that specifies only an OS profile.

1. In the Cloud Admin Portal, navigate to Manage > Resource Profiles.
2. In the Profiles list, select OS Profile and then open the profile.
3. On the Cloud Script OS Profile Mappings related list, click **New** and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blueprint (Optional)</td>
<td>Select a blueprint to limit the script to a specific resource block used in the blueprint.</td>
</tr>
<tr>
<td>Cloud script</td>
<td>Select a cloud script to map the OS profile to.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box if the cloud script should be run after the virtual machine is provisioned.</td>
</tr>
<tr>
<td>Application</td>
<td>Cloud Management is auto-selected.</td>
</tr>
<tr>
<td>OS profile</td>
<td>If you specify a blueprint, the cloud script is run when the blueprint is provisioned.</td>
</tr>
<tr>
<td>Resource Alias (Optional)</td>
<td>If you specify a resource alias for the blueprint, then the cloud script is executed when the blueprint with the specified resource alias is provisioned.</td>
</tr>
</tbody>
</table>

4. Right-click the form header and select **Save**. The **Name** attribute [scriptName] appears in the OS Profile Mapping Overrides list.

5. Optional: In the OS Profile Mapping Overrides list, enter an attribute name and value to use when the resource is provisioned.

Example: Post-provision cloud script
This example shows a cloud script that runs after a user provisions a virtual machine.

- Role required: cloud_admin
- OS resource profile that is mapped to an image template. The user provisions a resource using the profile. You must add basic authentication credentials to the image.
- Compute resource profile that is mapped to a hardware type.

In this example, you create a cloud init script, customize a resource block, and provision a VM from the Cloud User Portal.

1. Create a cloud script template and a cloud script that refers to the template.
2. Open the virtual server resource block, and then follow these steps:
   a) Set the Draft/Published switch to Draft.

   ![Virtual Server Interface](image.png)

   b) On the Operations tab, select **Virtual Server Interface** from the Interface list.
c) In the **Operation** field, verify that these operations exist:

- **Block Until Ready**
- **Execute Script**: This operation tells the blueprint to run a script.

![Operation field showing Block Until Ready and Execute Script options]

d) Click the **Input Parameters** tab to view the inputs for each operation.

e) On the **Steps** tab, add steps as needed.

f) For the **Script** parameter on each step, clear the **Script** check box and change the expression in the **Mapping** column to the following text:

```
$(CloudScript.scripts.scriptname)
```

where `scriptname` is the name of the script, not the script template, that you created earlier.
3. Publish the virtual server resource block, and use it in a blueprint that in turn is published to a catalog item.

4. Launch a new stack to provision a VM and specify the OS profile and Compute profile that you created as a prerequisite.

The Stack Status page runs through the **CreateNode**, **BlockUntilReady**, and **ExecuteShellScript** operations. The last operation kicked off the script you created. Validate a parameter value with the **AllowedPattern** property.

The **AllowedPattern** property in an Azure Resource Manager (ARM) and CloudFormation (CF) template performs input validation before sending a provisioning request.

Role required: sn.cmp.cloud_service_designer

You can provide rules to validate a parameter value before sending the provisioning request to the cloud. To be notified with an error message if the validation fails, specify the **ConstraintDescription** property in the template. You can customize the error message in the **ConstraintDescription** property to make the message clear and explicit. Usually, the CF templates already have the **AllowedPattern** and **ConstraintDescription** properties. If these properties are not present, you can add them as mentioned in the following procedure:

1. In the Cloud Admin Portal, navigate to **Design > Cloud Templates**.
2. Select a template or click **New** to create a new cloud template. If you select an existing template, rename the template.

3. In the **Body** text box, make the following changes in an ARM and a CF template.
   a) In an ARM template, make these changes:
      - In the parameters metadata section, enter `SNC::Parameter::Metadata`.
      - Add the **AllowedPattern** key and enter a valid value range.
      - Add the **ConstraintDescription** key and enter an error message.

        ```json
        "parameters": { 
            "newStorageAccountName": { 
                "type": "string", 
                "metadata": { 
                    "description": "Unique DNS Name for the Storage Account where the Virtual Machine's disks will be placed.", 
                    "SNC::Parameter::Metadata": { 
                        "allowedPattern": "[0-9a-z]{1,11}", 
                        "ConstraintDescription": "must be an alphanumeric " 
                    } 
                } 
            } 
        }
        ```

   b) In a CF template, make these changes:
      - Add the **AllowedPattern** key and enter a valid value range.
      - Add the **ConstraintDescription** key and enter an error message.

        ```json
        "Parameters" : { 
            "SSHLocation" : { 
                "Description" : "The IP address range that can be used to SSH to the EC2 instances", 
                "Type": "String", 
                "MinLength": "9", 
                "MaxLength": "18", 
                "Default": "0.0.0.0/0", 
                "AllowedPattern": "((\d{1,3})\.\d{1,3})\.(\d{1,3})/((\d{1,2}))", 
                "ConstraintDescription": "must be a valid IP CIDR range of the form x.x.x.x/x."
            } 
        }
        ```

4. Click **Submit**.

**Cloud Management governance**

Governance refers to the limitations that you can set on available cloud resources.

**Types of governance**

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Quotas</strong></td>
<td>Quotas are limitations or requirements for groups and users on a resource block. Use quotas to prevent wasteful resource usage in your cloud environment.</td>
</tr>
<tr>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Policies</td>
<td>A cloud policy can override a property value set by a user, create an approval task, reserve an IP address, pre-populate or hide form fields, execute custom scripts, call the Cloud API, or start or abort workflows. A cloud policy gives you system-wide control over approvals, resource operations, blueprint operations, or catalog item settings.</td>
</tr>
<tr>
<td>Pools</td>
<td>A resource pool is a query or script that filters a table. You configure a resource pool to limit the values that are available to users when they request a catalog item.</td>
</tr>
<tr>
<td>Permissions</td>
<td>Permissions are user group-level access rights to features in the Cloud Management application and to specific records in the instance, such as blueprints or cloud accounts.</td>
</tr>
</tbody>
</table>

*Permissions management for Cloud Management roles*

Permissions are user group-level access rights to features in the Cloud Management application and to specific records in the instance, such as blueprints or cloud accounts.

You can refine the actions that are allowed or prohibited for users based on user group. By default, each role includes access rights, but not all users with the same role can see or edit each others records. For example, by default users with the cloud_designer role have full read and write access to their own blueprints, but they do not have read or write access to blueprints created by other cloud designers. If you want them to have access, you must put the users into a group that has the cloud_designer role, then give that group read and write access to particular blueprints.

*Permission types*

**Access (Read)**

Users can see the record, but not edit or delete.

**Manage (Create & Update)**

Users can see the record and create and update new records in the same table, but cannot delete any records.

**Delete**

Users can delete specified records or all records in the table.

**Execute**

Users can execute an action on records in the table.

**All**

Users have permissions on all records in the table.

*Permissions suggestions*

Consider granting these permissions in your organization:
<table>
<thead>
<tr>
<th>Suggested group and role</th>
<th>Type of users in the group</th>
<th>Suggest permissions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog user group</td>
<td>Users who order similar items from the catalog in the Cloud User Portal.</td>
<td>Access (read) or Execute</td>
<td>Catalog items, even after they are published, cannot be seen by users in the Cloud User Portal until you grant a user group to which that user belongs read access to the items. Grant Access and Execute permissions to cloud users on the Blueprint Catalog Item table and the Cloud Account table.</td>
</tr>
<tr>
<td>Cloud user</td>
<td>[sn_cmp.cloud_service_user]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blueprint designers</td>
<td>Users who design blueprints.</td>
<td>Manage (Create and update)</td>
<td>Blueprint designers cannot see or edit other blueprints or catalog items by default. To collaborate or reuse existing blueprints and catalog items, blueprint designers need access to each others’ blueprints through the Manage permission.</td>
</tr>
<tr>
<td>Service Designer</td>
<td>[sn_cmp.cloud_service_designer]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cloud admins</td>
<td>Users who create and manage cloud accounts.</td>
<td>Manage (Create and update)</td>
<td>Cloud admins must map templates to appropriate resource profiles. To collaborate or reuse existing resource profiles and templates.</td>
</tr>
<tr>
<td>Cloud administrator</td>
<td>[sn_cmp.cloud_admin]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Assign a cloud permission

Assign a permission to refine the actions that are allowed or prohibited for users based on the user group they belong to.

- Role required: sn_cmp.cloud_governor
- The user group to which you want the permission applied.

1. Navigate to Cloud Management > Governance > Permission.
2. Fill out the form fields (see table).
Read permissions on cloud accounts

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target type</td>
<td>Select the cloud table in which the target record belongs.</td>
</tr>
<tr>
<td>All Entities</td>
<td>Select this option to apply the permission to all records in the table.</td>
</tr>
<tr>
<td>Permission</td>
<td>Select the permission type.</td>
</tr>
<tr>
<td>Target Entity</td>
<td>Select the record that the permission is based upon.</td>
</tr>
<tr>
<td>Group</td>
<td>Select the user group.</td>
</tr>
</tbody>
</table>

3. Click **Submit**.

**Policy governance in Cloud Management**

A cloud policy can override a property value set by a user, create an approval task, reserve an IP address, pre-populate or hide form fields, execute custom scripts, call the Cloud API, or start or abort workflows. A cloud policy gives you system-wide control over approvals, resource operations, blueprint operations, or catalog item settings.

A user requests a stack that triggers an approval policy.

1. On the Cloud User Portal, a user submits a request to provision a particular blueprint. The process of provisioning the blueprint is the trigger that causes the policy engine to apply an approval policy. A cloud approval policy specifies the users who must approve a specified cloud activity before the activity can proceed.

2. The policy engine determines that the request meets the condition specified in the rule for the policy. In this example, the condition evaluates to true whenever a particular blueprint is being provisioned.
3. Because the condition is met, the policy engine performs the action that is also specified in the rule. In this example, the action is to create an approval action for the manager.

4. While the approver (the manager) reviews the approval request, the user sees a 'waiting for approval' status message on the Cloud User Portal.

5. After the manager approves, the blueprint is provisioned.

About policy triggers

A trigger is an event — a user request or other activity — that triggers the policy engine. For example, the on Catalog item request end trigger fires after a user submits a request form. When you think about defining a policy, you might first consider how the policy will be triggered. Examples:

- A user requests a Stop operation on a virtual server (the on Stack resource operation trigger fires)
- A resource reaches the end of its lease (the on Lease End trigger fires)
- A user requests a particular stack (the on Blueprint provision trigger fires)

You typically refer to a policy by the name of the trigger for the policy. For example, you might refer to a policy that is triggered by the on Lease end trigger as a "Lease end policy." For more detail on the types of trigger that you can implement, see Triggers for cloud policies.

How policies work

For each policy, you typically configure a trigger that starts the policy engine. The policy engine enforces the policy by running all rules that are defined for the policy. A policy rule is a collection of conditions and actions. If all conditions evaluate to true, the policy engine performs the actions. If any condition evaluates to false, the policy engine does not perform the actions.
• Some policy types apply only to particular types of cloud operation, like start, stop, provision, or deprovision, or to a particular target, like ‘on blueprint123 provision operation’ or ‘on catalog item ABC launch’.

• You can configure a policy that does not specify a target, for example, ‘on any blueprint provision operation’ or ‘on any catalog item launch’. Errors on policies that apply to any object are ignored.

• If multiple policies apply, you can specify the order that the policies are applied (with an exception that is described in a following section).

• Policies can work with dynamic forms to allow you to show or hide form fields from end users. Users see only information that you decide they need for them to understand and complete their tasks.

Policy groups

A cloud policy group is a container for related policies. Consider grouping policies that are often used together or should be considered together. Grouping policies can help you to apply policies consistently across your organization.

Order of execution when multiple policies apply

Order of execution when multiple "approval" policies apply:

When multiple "approval" policies apply, the policies are applied in the following order. (The approval policies are on Blueprint provision (approval), on Stack operation (approval), on Stack resource operation (approval), and on Task remediation):

1. Only the first successful approval policy is applied and no other approval policies are applied.
2. If the applied approval policy has multiple rules, only the first successful rule is used.
3. If a rule has multiple actions, only the first successful action is performed.
4. If the applied approval policy includes both a custom approval and a Service Now approval, only the custom approval process is performed.

Order of execution for all other policy types:

• Policies are applied in the order that is specified by the Order of Execution property setting.
• If multiple policies have the same Order of Execution setting, then order is not guaranteed.

About operations in policies

Triggers are often based on user requests and the operations (start, stop, provision, or deprovision) that can run on a blueprint, a catalog item, a resource, or a stack. Some trigger types do not specify a cloud operation. For example, the on Lease End trigger fires independently of any operation.
About policy rules

A policy rule is a collection of conditions and actions. If all conditions evaluate to true, the policy engine performs the actions. If any condition evaluates to false, the policy engine does not perform the actions.

- Conditions: Conditions can consider request form data, resource activity, or user activity. Examples:
  - Is the size of the requested CPU greater than 32?
  - Is this a Stop operation?
  - Is the lease for this resource ending in the next 7 days?

- Actions: If all conditions evaluate to true, the policy engine runs the actions that the rule specifies. Expressions in policy actions can set or override values. Examples:
  - Override the value that the user specified by changing the CPU size to 16 and then start the approval process. (because the condition was met that the requested CPU is greater than 32).
  - Create an approval task for the manager. (because the condition was met that a Stop operation was requested).
  - Send notifications to every user in the ABC group. (because the condition was met that the lease ends in 7 days).

Policy action scripts

- Use policy action scripts to get, update, or set values in cloud requests.
- Your instance tracks tagged resources for billing and reporting. Policy action scripts can add and modify resource tags.

Cloud policy example

A base system cloud policy, Lease End ServiceNow, uses the on Lease end trigger that fires when a virtual resource is near the lease end date. If the conditions in the policy rule are met, the policy engine sends a notification to the owner of the resource and performs operations on the resource.

Role required: sn_cmp.cloud_governor or admin

In this example, you work with the Lease End ServiceNow policy. The policy performs the following actions when the conditions in the policy rule are met:

- Send a notification to the owner of a cloud resource when the resource lease is about to expire.
- Run a Stop operation on the resource.
- Run a Deprovision operation on the resource.

You can change any base-system policy by modifying settings or making the policy active or inactive.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Search for and open the Lease End ServiceNow policy.
3. Notice that the trigger is on Lease end, which means that the policy is triggered when the lease for the resource is near expiration.
4. Sort the Rules related list by **Order of Execution**. The rule with the lowest number runs first.
Notice that a notification rule runs first, followed by the Stop and Deprovision operations on the resource.

5. Click the Notify about upcoming lease end rule.
6. Notice that the rule specifies a condition: If it is one day before the end date, return true.

7. In the Policy Rule Actions related list, click the (empty) Name of the action.
8. On the Update Action popup, enter an Action Name (the value is required). Notice the value of the Notification (you will use it in the next step), and then close the popup.
9. Review the notification message other settings:
   a) On the instance (not the Cloud Admin Portal), navigate to **System Notification > Email > Notifications**.
   b) Search for and open the notification that is specified in the action by entering the filter text *Upcoming lease end* for **Name**:
   
   ![Notification Search](image)

   c) Click **Upcoming lease end** to open the notification form.
   
   Notice that the **Scheduled Operations (sn CMP scheduled_operation)** table is specified.

   **Note:** You base all notifications for Cloud Management on this table.
d) On the **When to send** tab, notice the conditions that apply:
<table>
<thead>
<tr>
<th>Condition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type is Notification</td>
<td>Specifies that the <strong>Operation Type</strong> field on the Scheduled Operations table must be <strong>Notification</strong>.</td>
</tr>
<tr>
<td>Status</td>
<td>Specifies that the system should trigger the notification when the status of the scheduled operation record is in progress, rather than waiting until it is complete.</td>
</tr>
<tr>
<td>Notification</td>
<td>Specifies that the <strong>Upcoming lease end</strong> notification must be specified.</td>
</tr>
</tbody>
</table>

**Fastpath:** When you create a custom notification, use the default conditions, but change the **Notification** condition to the name of your notification.

e) Click the **Who will receive** tab.

Notice that the **Users/Groups in fields** value is set to **Target.Assigned to, TargetOwned by**. You should use these settings to send the notification to the owner of the virtual resource.

10. In the Cloud Admin Portal, open the Lease End ServiceNow policy again, and then open the **Stop on lease end** rule.

![Image of Cloud Admin Portal with Lease End ServiceNow policy]

The rule has two actions — one action to send a notification and the other action to run the **Stop** operation on the virtual resource.

11. Click the (empty) **Action Name** of the action for the **Stop** operation.
12. On the Update Action popup, notice the settings:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy and Rule Name</td>
<td>The policy and the rule to which this action belongs are specified in these fields by default.</td>
</tr>
<tr>
<td>Action Type</td>
<td>Run An Operation tells the policy engine to trigger a standard operation (Start, Stop, Deprovision, and so on) on the resource.</td>
</tr>
<tr>
<td>Blueprint Operation</td>
<td>The system runs the STOP operation on the resource.</td>
</tr>
</tbody>
</table>

The other rule, **Deprovision after lease end**, performs the Deprovision operation on the resource so that it does not stay active after the lease end date.

This example uses the settings available through the form configuration. To see example scripts that you could use in addition, see [Create a policy action script](#).

Triggers for cloud policies
A trigger is an event — a user request or other activity — that triggers the policy engine. For example, the on Catalog item request end trigger fires after a user submits a request form. When the trigger for a policy fires, the policy engine tests the conditions that are specified in the policy rule and, if the conditions are met, performs the actions that are specified in the rule.

### About triggers

- You typically refer to a policy by the name of the trigger for the policy. For example, you might refer to a policy that is triggered by the on Lease end trigger as a “Lease end policy.”
- Triggers are often based on user requests and the operations (start, stop, provision, or deprovision) that can run on a blueprint, a catalog item, a resource, or a stack. Some trigger types do not specify a cloud operation. For example, the on Lease End trigger fires independently of any operation.
- To optimize performance, limit the number of policies with general triggers like the on Catalog item triggers.
- A trigger that does not specify a target (a blueprint, catalog item, stack, or resource) is always executed. To optimize performance, therefore, minimize the use of such policies.
### Policy triggers

Types of policy trigger and the actions that are available for the associated policy rules

<table>
<thead>
<tr>
<th>Trigger name and actions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>on Blueprint provision</td>
<td>The <em>on Blueprint provision</em> trigger fires after execution of on Catalog item request start policies. A policy that is triggered by the <em>on Blueprint provision</em> trigger can run a script, override a user-requested attribute value, or abort and send a message about the provision operation. Use this trigger to override a value that the user enters. For example, when a user chooses a value for an attribute like the stack name, a policy with this trigger can change the stack name. In addition, another action can change the name again when the user finally provisions the resource. The user does not see the final value on the catalog item form because the change is made at provision time.</td>
</tr>
<tr>
<td>Actions:</td>
<td>- Execute a script&lt;br&gt;- Property override&lt;br&gt;- Abort process</td>
</tr>
</tbody>
</table>

*Create an action for an ‘on Blueprint provision’ policy*
<table>
<thead>
<tr>
<th>Trigger name and actions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approval triggers</td>
<td>A cloud approval policy specifies the users who must approve a specified cloud activity before the activity can proceed. Approvers can include the manager of the user making a request, a specified user or group, or users with a specified role. You can specify multiple approvers. Approvals occur in the order that you specify. A policy that is triggered by one of the approval triggers can start approval workflows. The targeted approval policies complement the base-system approval operations.</td>
</tr>
<tr>
<td>- on Blueprint provision (approval): Triggered when the workflow begins to provision a blueprint. The trigger is invoked before the blueprint is provisioned. Because the provisioning process can alter request data (and possibly change costs), however, approval processes run after the blueprint is provisioned.</td>
<td>Note: The approval process is performed after properties are set because property values that were overridden could change costs.</td>
</tr>
<tr>
<td>- on Stack operation (approval): Triggered during any stack operation on the Cloud User Portal.</td>
<td>on Blueprint provision (approval) is applied before the blueprint is provisioned. Because the provisioning process can alter request data (and possibly change costs), approval processes run after the blueprint is provisioned.</td>
</tr>
<tr>
<td>- on Stack resource operation (approval): Triggered during any resource operation (start, stop, provision, and so on) on the Cloud User Portal.</td>
<td>Use on Stack operation (approval) to run an approval workflow when an operation is performed on a stack. By default, a change request is generated when an operation is performed on a stack, but it does not require an approval. This trigger can launch a mandatory approval.</td>
</tr>
<tr>
<td>- on Task remediation: Triggered when a user resubmits a failed request.</td>
<td>Use on Stack resource operation (approval) to run an approval workflow when an operation is performed on a single resource that is part of a stack. By default, a change request is generated when an operation is performed on a stack, but it does not require an approval. This trigger can launch a mandatory approval.</td>
</tr>
<tr>
<td>Actions:</td>
<td>A policy that is triggered by the on Task Remediation trigger can start approval workflows.</td>
</tr>
<tr>
<td>- ServiceNow Approval</td>
<td>Create an action for an 'approval' policy</td>
</tr>
<tr>
<td>- Custom Approval</td>
<td></td>
</tr>
</tbody>
</table>

Create an action for an ‘approval’ policy.
<table>
<thead>
<tr>
<th>Trigger name and actions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>on Catalog item launch</strong></td>
<td>The <em>on Catalog item launch</em> trigger fires when an order form (stack request form) is launched for a catalog item. A policy that is triggered by the <em>on Catalog item launch</em> trigger can run a script or override a user-requested value (text values only). Use this trigger to control what the user sees in the form when it first opens in the Cloud User Portal. For example, you can override a default value that first appears to the user. The user can see this value on the catalog item form. When a value is overwritten by both a policy rule and a form rule, the value in the form rule is used. Create an action for an <em>on Catalog item launch</em> policy.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>· Execute a script</td>
<td></td>
</tr>
<tr>
<td>· Property override</td>
<td></td>
</tr>
<tr>
<td><strong>on Catalog item request start</strong></td>
<td>The <em>on Catalog item request start</em> trigger fires after a user opens a request form. A policy that is triggered by the <em>on Catalog item request start</em> or <em>on Catalog item request end</em> trigger can run a script or execute a workflow. You can use this trigger to run a custom script or workflow to fulfill enterprise processes like custom approval before the catalog item request is processed. Create an action for an <em>on Catalog item request start/end</em> policy.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>· Execute a script</td>
<td></td>
</tr>
<tr>
<td>· Execute a workflow</td>
<td></td>
</tr>
<tr>
<td><strong>on Catalog item request end</strong></td>
<td>The <em>on Catalog item request end</em> trigger fires after a user submits a request form. A policy that is triggered by the <em>on Catalog item request start</em> or <em>on Catalog item request end</em> trigger can run a script or execute a workflow. Use this trigger to launch a workflow after a catalog item request is processed. Consider this trigger a post-provisioning step. For example, you could launch a workflow to install MySQL on the provisioned resource. Create an action for an <em>on Catalog item request start/end</em> policy.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>· Execute a script</td>
<td></td>
</tr>
<tr>
<td>· Execute a workflow</td>
<td></td>
</tr>
<tr>
<td><strong>on Lease end</strong></td>
<td>A policy that is triggered by the <em>on Lease end</em> trigger can send a notification or perform a Start, Stop, or Deprovision life cycle operation. Create an action for an <em>on Lease end</em> policy.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>· Run an Operation</td>
<td></td>
</tr>
<tr>
<td>· Send a Notification</td>
<td></td>
</tr>
<tr>
<td>Trigger name and actions</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>on Resource operation launch</td>
<td>The on Resource operation launch trigger fires before the catalog for a resource operation is loaded from the Cloud User Portal. A policy that is triggered by the on Resource operation launch trigger can run a script or can override a user-requested value (text values only). When a value is overwritten by both a policy rule and a form rule, the value in the form rule is used.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>• Execute a script</td>
<td></td>
</tr>
<tr>
<td>• Property override</td>
<td></td>
</tr>
<tr>
<td>on Resource operation request start</td>
<td>The on Resource operation request start trigger fires after a user submits a resource operation request (Start, Stop, Deprovision). A policy that is triggered by the on Resource operation request start or on Resource operation request end trigger can run a script or override a user-requested attribute value.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>• Execute a script</td>
<td></td>
</tr>
<tr>
<td>• Execute a workflow</td>
<td></td>
</tr>
<tr>
<td>on Resource operation request end</td>
<td>The on Resource operation request end trigger fires before completion of a life cycle operation on a resource (Start, Stop, Deprovision). A policy that is triggered by the on Resource operation request start or on Resource operation request end trigger can run a script or override a user-requested attribute value.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>• Execute a script</td>
<td></td>
</tr>
<tr>
<td>• Execute a workflow</td>
<td></td>
</tr>
<tr>
<td>on Resource operation</td>
<td>The on Resource operation trigger fires during the Orchestration process when a user performs a Start, Stop, or Deprovision life cycle operation on a specific resource. A policy that is triggered by the on Resource operation trigger can override a user-requested attribute value, run a script, call a Cloud API, or perform an IP address management operation.</td>
</tr>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>• Property override</td>
<td></td>
</tr>
<tr>
<td>• Execute a script</td>
<td></td>
</tr>
<tr>
<td>• Call Cloud API</td>
<td></td>
</tr>
<tr>
<td>• IP Address Management</td>
<td></td>
</tr>
</tbody>
</table>

Create a cloud policy

A cloud policy can override a property value set by a user, create an approval task, reserve an IP address, pre-populate or hide form fields, execute custom scripts, call the Cloud API, or start or abort workflows. A cloud policy gives you system-wide control over approvals, resource operations, blueprint operations, or catalog item settings.

- Role required: sn_cmp.cloud_governor or admin
- Optional: [Create one or more cloud policy groups](#)
This procedure describes every policy type except approval policies (on Blueprint provision (approval), on Stack operation (approval), on Stack resource operation (approval), and on Task remediation). See Create a cloud approval policy for instructions on creating a policy with an approval trigger.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Click New and then fill in the form.
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Trigger</td>
<td>Select a trigger that specifies when the policy should be applied. For example, the <strong>on Lease end</strong> trigger applies the policy when the lease for a stack ends. See <a href="#">Triggers for cloud policies</a>.</td>
</tr>
<tr>
<td>Resource Block</td>
<td>Select the resource block that the policy applies to.</td>
</tr>
</tbody>
</table>
| Blueprint | Select the blueprint that the policy applies to.  
  - If no blueprint is specified, then the policy applies for every blueprint. This condition can decrease performance.  
  - You can assign multiple policies to a blueprint. Multiple policies running simultaneously, however, might decrease performance. Publish a policy only when it should be enforced. |
| Operation | Select the operation that the policy applies to. For example, a policy can apply to the Deprovision operation only or to all operations on the blueprint or catalog item.  
  - **All**: Any operation executes.  
  - **Start**: The resource starts.  
  - **Stop**: The resource stops.  
  - **Provision**: The resource is provisioned.  
  - **Deprovision**: The resource is no longer available to users.  
  - **Execute Script**: A script runs on the resource. |
| Moment | Specify when the policy should be enforced:  
  - **Pre-operation**: Before the specified operation starts.  
  - **Post-operation**: After the specified operation finishes. |

Note: If you are integrating with Infoblox, use **Pre-operation** for a vSphere virtual machine. Use **Post-operation** for AWS and Azure clouds because AWS and Azure control the allocation of IP addresses. You can register the IP address that is provided with Infoblox.
### ServiceNow New York IT Operations Management

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog item (appears when an on Catalog item launch, on Catalog item request start, or on Catalog item request end trigger is selected)</td>
<td>Select the catalog item that the policy applies to.</td>
</tr>
<tr>
<td>Start Date / End Date</td>
<td>Specify the start date when the policy should be considered and the end date when the policy should no longer be considered.</td>
</tr>
<tr>
<td>Order of Execution</td>
<td>Specify a number that represents the order in which the policy is applied. A policy with a lower number runs before a policy with a higher number.</td>
</tr>
</tbody>
</table>

For example, a policy with **Order** of **100** runs before a policy with an **Order** of **200**.

**Note:** The **Order of Execution** property does not apply for on Blueprint provision (approval), on Stack operation (approval), on Stack resource operation (approval), and on Task remediation policies. See [Create a cloud approval policy](#) for details.

<table>
<thead>
<tr>
<th>Status</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>· <strong>Published</strong> policies are enforced. You cannot edit a policy in the <strong>Published</strong> state. To edit a policy that is in the <strong>Published</strong> state, click <strong>Draft</strong> on the form header.</td>
<td></td>
</tr>
<tr>
<td>· <strong>Draft</strong> policies are not enforced. To enforce a policy, click <strong>Publish</strong> on the form header.</td>
<td></td>
</tr>
</tbody>
</table>

3. Right-click in the header and select **Save**.

**Configure a cloud policy rule** for the policy.

Configure a cloud policy rule

A policy rule is a collection of conditions and actions. If all conditions evaluate to true, the policy engine performs the actions. If any condition evaluates to false, the policy engine does not perform the actions.

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups.

- A policy can include multiple rules. You can configure the order that rules are applied.
- Every rule is executed to completion.
- Any rule error stops execution. Exception: Policies that do not specify an operation.

1. In the Cloud Admin Portal, navigate to **Govern > Policies**.
2. Open a cloud policy. Set the policy to the **Draft** state if needed.
3. On the Rules related list, click **New** and then enter a descriptive **Name** for the rule.
- The rule name cannot be the same as the policy name.
- The rule name should not start with a number.

4. Using the base-system condition builder, specify the **Conditions** that must be met for the policy engine to perform the actions.

The following example rule is applied for a blueprint provision approval policy. The condition evaluates to true if the **OS profile** field on the request form has the value “Windows”.
### Field

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Criteria type)</td>
<td>Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Request Form</strong>: Base the condition on a property (attribute) value in the request form in the Cloud User Portal. You specify which property to test. In the example, the condition tests whether the OS profile field on the request form has the value “Windows.”</td>
</tr>
<tr>
<td></td>
<td>• <strong>User</strong>: Base the condition on a user role or on the group that the user belongs to. For example, to create a condition that applies when the user belongs to the Marketing group, select:</td>
</tr>
</tbody>
</table>
|                                            |   • (Criteria Type): User  
|                                            |   • (User Entity): Group  
|                                            |   • (Relational Operator): Equals  
|                                            |   • (Group): Marketing |

**Note:** You use different criteria to build a condition for policies that are triggered by the **on Lease end** trigger. See the example that appears after the table.

<table>
<thead>
<tr>
<th>(Relational Operator)</th>
<th>Select an appropriate operator.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Value)</td>
<td>Depending on the selected (Criteria type), enter or select a value.</td>
</tr>
<tr>
<td></td>
<td>• For <strong>Request Form</strong>, enter a value for the attribute.</td>
</tr>
<tr>
<td></td>
<td>• For <strong>User &gt; Group</strong>, select a group.</td>
</tr>
<tr>
<td></td>
<td>• For <strong>User &gt; Role</strong>, select a role.</td>
</tr>
</tbody>
</table>

| (Logical operator)                         | When defining multiple conditions, select a logical operator **OR** or **AND** that specifies the logical relationship with the next condition. |

5. **Click Submit.**

For policies that are triggered by the **on Lease end** trigger, use the base-system date condition builder to configure the condition to test. In addition to the condition, the following example shows the actions that the policy engine performs if the condition is met.
Create the policy actions that should run when all conditions of the rule evaluate to true.

Using expressions in Cloud Management

Expressions in policy actions can set or override values. Expressions in blueprints can access attributes of resources and can map values to request form fields. Expressions are available in resource blocks, blueprints, policies, and anywhere that Cloud Management allows scripts.
Using expressions

Expressions can hold the values for information that is dynamically generated in the system, such as the values for the selections users make on the Cloud User Portal. Expressions are available in resource blocks, blueprints, policies, and anywhere that Cloud Management allows scripts.

Using expressions in resource blocks

Each resource block operation uses an expression to hold a value for each parameter. The expression can use hard-coded values, data from the stack that the user provisions in the Cloud User Portal, data in the CMDB, and data that is derived from scripts. By default, Cloud Management generates a set of parameters and their expressions for each resource block operation.

For example, the parameter **Location**, which holds the datacenter that a resource belongs to, uses the expression `${parameter.Location}`
Blueprints can use expressions to map input parameters for each operation on a resource block. By default, the blueprint displays the same parameters and values that are specified in the resource block.

You can access blueprint attributes with expressions. The expression in the **Mapping** column defines blueprint attributes for each operation in a step. For example, the **Location** attribute, which appears on the stack request form, is accessible through the \(\{\text{parameter.Location}\}\) expression. The Location parameter with the \(\{\text{parameter.Location}\}\) value appears on the **Inputs** tab when you select a blueprint operation.

Using expressions in policy actions

Policy actions can use expressions to override parameter values that users submit in a request form on the Cloud User Portal. You can also access and override user data in stack operations. For example, use the following expression to rename a stack: \(\text{formData.Stackname} = \text{"your-naming-convention"};\).
You can access blueprint attributes with expressions. The expression in the **Mapping** column defines blueprint attributes for each operation in a step. For example, the **Location** attribute, which appears on the request form for a virtual machine, is accessed in the following expression: 

```
{$parameter.Location}
```

### Expression types

#### Definitional expressions

Definitional expressions are a form of early binding (or static binding). These bindings refer to compile-time binding and are evaluated when the user submits the stack request form (when stack or resource provisioning starts). Definitional expressions do not have access to the data that is generated during Orchestration. Definitional expressions are enclosed in curly braces. For example: 

```
${parameter.CloudAccount}
```

#### Runtime expressions

Runtime expressions are a form of late binding (or dynamic binding). These bindings refer to runtime binding and have access to the data that is created during Orchestration (for example, stack items). Runtime expressions are evaluated when called during an Orchestration run.
Runtime expressions are typically used for dot-walking to data in tables. Runtime expressions are enclosed in parentheses. For example: $(Stack.items[VM1].attributes[node_id])

**Definitional/Runtime expressions**

Some expressions include both definitional and runtime expressions.

**Complex expressions**

You can nest expressions of any type within other expressions.

### Definition expression syntax and examples

Allowable syntax uses a dollar sign and curly braces. These types are available:

- `${parameter.}`. Use this kind of expression to retrieve values from input attributes of processes like blueprint provisions. Data is not fetched from tables.
- `${Stack.items[]}`. Use this syntax to access attributes of specific items in a stack.
- `${randomNumber}`. Use this syntax to generate a random number. For example, you can provision a VM with a random node name or stack name.

#### Definition expression examples

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>${parameter.BillingCode}</code></td>
<td>This expression shows how a billing code parameter appears in a blueprint.</td>
</tr>
<tr>
<td><code>${parameter.formData.CatalogAttributeType}</code></td>
<td>This expression takes a catalog attribute that is submitted by a user while the user fills out the form for a catalog item in the Cloud User Portal.</td>
</tr>
<tr>
<td><code>${parameter.userData.userId}</code></td>
<td>This expression takes the ID of a user while the user fills out the form for a catalog item in the Cloud User Portal.</td>
</tr>
<tr>
<td><code>${Stack.items[Virtual Server].attributes(sys_id)]</code></td>
<td>This expression gets the sys_id of a virtual server that is apart of a stack. Virtual Server is the alias of a resource block used in the stack.</td>
</tr>
</tbody>
</table>

### Runtime expression syntax and examples

Allowable syntax uses a dollar sign and parenthesis. These types are available:

- `$({tablename})` where tablename is a table in the system, usually a CI table in the CMDB. Use this syntax to access values of fields in the table.
- `$({Stack.items[]})`. Use this syntax to access attributes of specific items in a stack.

#### Runtime expression examples

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>$({ci.cmdb_ci_cloud_subnet[ sys_id=12231231231231231231].cidr)</code></td>
<td>This expression dot walks to the Cloud Subnet table, finds the specific record with the given sys_id, and takes the value from the cidr column.</td>
</tr>
<tr>
<td>Example</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>$(ci.sn_cmp_ip_pool[subnet=${parameter.formData.Subnet Id}])</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>This expression combines runtime and definition type expressions. The equal sign = is used to evaluate a value for a match. The expression dot-walks to the IP Pool table and looks for the subnet that has the subnet ID that the user submitted.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(ci.cmdb_ci_cloud_subnet[${parameter.formData.SubnetId}].cidr)</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>This expression combines runtime and definition type expressions. The expression takes the value of the <em>cidr</em> field from a subnet that the user chose during provisioning. The square brackets [ ] indicate that the expression dot-walks to the Cloud Subnet table and then looks at the subnet value that the user submitted on a blueprint provision. The expression grabs the <em>cidr</em> field value and then walks to the value in the <em>cidr</em> field of the <em>sys_id</em> of the subnet.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(Stack.items[Virtual Server].attributes(sys_id))</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>As in the definition expression example, this expression takes the <em>sys_id</em> of a virtual server that is apart of a stack.</td>
</tr>
</tbody>
</table>

**Example expression**

```$\{(ci.cmdb_ci_nic[$(Script:CMPVMNICs.getNICs[arg=$(Stack.items[Virtual Server].attributes[sys_id])])].private_ip)\}$
```

- **$ci:** Runtime expression to retrieve data from table.
- **cmdb_ci_nic:** CI for NIC (Network Interface Card).
- **$(Script):** Script-based expression.
- **CMPVMNICs:** Script include.
- **getNICs:** Function inside a script include.
- **arg:** Arguments to script include function. Arguments are separated by ",," when there are multiple attributes.
- **$(Stack.items):** Runtime stack expression to retrieve stack item from a stack. Argument is the alias specified in blueprint.
- **Virtual Server:** Alias of resource used in blueprint.
- **$(Stack.items(Virtual Server).attributes(sys_id))**: Retrieve *sys_id* of stack item ("Virtual Server") resource instance in a stack.
- **private_ip:** Attribute from cmdb_ci_nic. Replace with public IP if required.
Expressions

Simple parameter mapping expression
This type of expression retrieves values from input attributes of processes, such as blueprint operations, resource blocks, and policies. Data is not fetched from tables. Maps values from one layer to another, such as from a Blueprint to a Resource to the Cloud API.

- Type: definitional
- Syntax: \${parameter.}
- Examples:
  - \${parameter.BillingCode} returns the billing code.
  - \${parameter.formData.CatalogAttributeType} takes a catalog attribute that is submitted on the request form in the Cloud User Portal.
  - \${parameter.userData.userId} takes the ID of a user working on the request form in the Cloud User Portal.

Stack item expression
A CI instance in the CMDB represents each stack item. Use Stack Item expressions to look up first-level properties on the CI that back the stack item or on the stack item itself.

- Type: definitional/runtime
- Syntax: \${Stack.items[]} or $(Stack.items[])
- Examples:
  - \${Stack.items[VirtualServer1].attributes[sys_id]} is a definitional type expression that gets the sys_id of a virtual server in a stack. VirtualServer1 is the alias of a resource block that is used in the stack.
  - $(Stack.items[VirtualServer2].attributes[sys_id]) is a runtime type expression that takes the sys_id of a virtual server that is a part of a stack.
  - \${Stack.items[vm1].attributes[node_id]} reads the node_id attribute of the CI that was created for the VM. vm1 is the stack item name (or the alias of the resource in the blueprint).
  - $(Stack.items[vm1].status) reads the status of the stack item.
Property override expressions in policies

In policies, you can override properties by pulling a value from the system or by using a random number. You can use data from both the forms in the Cloud User Portal and from the user who performed the operation on the form.

- **Type:** definitional/runtime
- **Syntax:** `${parameter.formData.xyz}` or `${this.parameter.userData.xyz}`
- **Examples:**

  The following value can set the stack name to `Stack_Bob.Smith@company.com`:

  ```
  Stack_ ${parameter.userData.userId}
  ```

  Set stack or table values using runtime expressions by taking the subnet from the IP Pools table:

  ```
  $(ci.sn_cmp_ip_pool[subnet=${parameter.formData.Subnet Id}])
  ```

  In scripts, you can assign values as follows:

  ```
  formData.App_Server_NodeName = "MyNodeName";
  this.parameter.userData.userId == 'servicenowuserId';
  ```

- **See also** Create a policy action script.

Script expression

In the example, `VMProperties` is a script include with a function called `getIP`. A script expression is also an example of a complex expression (the expression is nested).

- **Type:** runtime
- **Syntax:** `$(Script:scriptName.function[])`
- **Examples:**

  ```
  $(Script:VMPropertiesUtil.getIP[ arg=
  $(Stack.items[VM1].attributes[object_id])])
  ```

  **Expression for Private IP:**

  ```
  $(ci.cmdb_ci_nic[$(Script:CMPVMNICs.getNICs[arg=
  $(Stack.items[Virtual Server].attributes[sys_id])])].private_ip)
  ```

  **Expression for Public IP:**

  ```
  $(ci.cmdb_ci_nic[$(Script:CMPVMNICs.getNICs[arg=
  $(Stack.items[Virtual Server].attributes[sys_id])])].public_ip)
  ```

  **Expression to get the credential alias:**

  ```
  $(Script:CMPVMUtils.getCredentialAlias[arg=
  $(Stack.items[Virtual Server].attributes[sys_id])])
  ```

  **Expression to get the IP address:**

  ```
  $(Script:CMPVMUtils.getReachableIp[arg=
  $(Stack.items[Virtual Server].attributes[sys_id])])
  ```
### CI lookup expression
Accesses values of fields in a table, usually a CI table in the CMDB.

- **Type:** definitional/runtime
- **Syntax:** $(ci.tableName)
- **Examples:**
  
  ```
  $(ci.sn_cmp _ip_pool[subnet=${parameter.formData.Subnet Id}])
  ```

  This combines runtime and definition type expressions. The = operator evaluates a value for a match. The expression dot-walks to the IP Pool table and looks for the subnet that has the subnet ID that the user submitted.

  ```
  $(ci.cmdb_ci_cloud_subnet[${parameter.formData.SubnetId}].cidr)
  ```

  The expression dot-walks to the Cloud Subnet table, looks at the subnet value that the user submitted on a blueprint provision, extracts the cidr field value, and then walks to the value in the cidr field of the sys_id of the subnet.

### Random number expression
Generates a random number. For example, you can provision a VM with a random node name or stack name.

- **Type:** runtime
- **Syntax and example:** `${randomNumber}

### Scratchpad expression/Resource operation output attribute expression
Reads output attributes from one operation into another operation.

- **Type:** runtime
- **Syntax:** varies
- **Example:** To set/expose outputs from one operation:
  ```
  $(Compute Interface.CreateNode.Output.resp.nodeId)
  ```

  To read the output attributes (where VM1.Provision is the operation whose output attributes are read):
  ```
  $(Outputs[VM1.Provision].NodeId)
  ```

### Conditions in expressions
You can use conditional expressions in blueprint steps and resource operation steps to conditionally execute or skip the step. The conditional expressions are Javascript expressions and they support expression substitutions.

- **Type:** definitional/runtime
- **Syntax:** varies
- **Examples:**
  ```
  '${{parameter.CloudAccount}}'=='Amazon Cloud Account'
  ```
  ```
  '${{Stack.items[vm1].attributes[node_id]}}'=='VM1'
  ```
Complex expression
You can nest expressions. In the example, ${parameter.ServerID} maps the sys_id of the CI and is replaced before the outer expression is consumed.

- Type: any
- Syntax: varies
- Example: ${ci.cmdb_ci_vm_instance[${parameter.ServerID}].name}

Order context
This expression is useful for life cycle operations and enables you to dot-walk on the order attributes as part of the sn_cmp_order table.

- Type: runtime
- Syntax: $(context.order.column_name)
- Examples:
  - Request Item: $(context.order.sc_req_item)
  - Dot-walk on request item object: $(context.order.sc_req_item.number)
  - Mixed Expression (Constant + Expression): "ram $(context.order.sc_req_item.number)"

Create an action for an 'on Blueprint provision' policy
The on Blueprint provision trigger fires after execution of on Catalog item request start policies. A policy that is triggered by the on Blueprint provision trigger can run a script, override a user-requested attribute value, or abort and send a message about the provision operation.

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups. Configure a cloud policy rule and associated conditions.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform, enter a unique and meaningful Action Name, and then fill in the form for the action.
Create Policy Rule Action

**Execute a Script**

This action will execute a ServiceNow script on the instance. All the operation parameters are available as inputs to the script.

**Property Override**

This action will set the value of a given parameter of the operation.

**Abort Process**

This action will stop the workflow which drives the overall orchestration.

Settings for the 'Execute a Script' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>Select a category.</td>
</tr>
<tr>
<td>Action Script Name</td>
<td>Specify a unique and meaningful name for the script.</td>
</tr>
<tr>
<td>Action Script</td>
<td>Create the script in the text box.</td>
</tr>
<tr>
<td></td>
<td>See <a href="#">Create a policy action script</a> for details.</td>
</tr>
</tbody>
</table>

Settings for the 'Property Override' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
<td>Specify the name of the property (attribute) on the user request form to override.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Value</td>
<td>Enter a value that overrides the value in the Property field. You can override text values only. You can specify a static value, an expression, or both. The example action, named SetTheCostCenter, specifies the value Marketing for the CostCenter property.</td>
</tr>
</tbody>
</table>

**Property Override**

- **Policy**: ProvisionActions
- **Rule Name**: FirstRuleProvision
- **Action Type**: Property Override
- **Action Name**: SetTheCostCenter
- **Property**: CostCenter
- **Value**: Marketing
- **Is Script Based**: [ ]

**Note:**
When a value is overwritten by both a policy rule and a form rule, the value in the form rule is used.

Expressions can perform the following actions (see Using expressions in Cloud Management for details):

- Set form data values using definition expressions. For example: ${parameter.formData.CatalogAttributeType}
- Assign user data values using definition expressions. For example: ${parameter.userData.userId}
- Set stack or table values using runtime expressions. For example: $(ci.sn_cmp_ip_pool[subnet=${parameter.formData.SubnetId}]) takes the subnet from the IP Pools table.
- Associate a random number with a field using static expressions. Use: ${randomNumber}
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is Script Based</td>
<td>Select the check box to display the <strong>Script</strong> text box and then specify the script. You can use the following example script snippet to override a stack name. The <strong>function( formData)</strong> section of the script modifies the values for fields on the form. MyStack is the stack name in this example.</td>
</tr>
</tbody>
</table>
|                       | ```javascript
customScript : function( formData){
  // Manipulation of form parameter is only supported here.
  // Change in any other attributes will be ignored
  // data available for manipulation are
  // Form Data
  // Ex. StackName can be accessed through formData.StackName
  formData.StackName = "MyStack";
  // User Data - Ex. User Id can be accessed through this.parameters.userData
  if(this.parameter.userData.userId == 'servicenowuserId')
  formData.StackName = "MyStack";
  return formData;
}
``` |

**Setting for the 'Abort Process' action**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
<td>Enter the message to present to the requester when the process aborts.</td>
</tr>
</tbody>
</table>

Create an action for an 'approval' policy

A policy that is triggered by one of the approval triggers can start approval workflows. The approval triggers are (on Blueprint provision (approval), on Stack operation (approval), on Stack resource operation (approval), and on Task remediation.

- **Role required:** `sn_cmp.cloud_governor` or `admin`
- **Optional:** Create one or more cloud policy groups.
Configure a cloud policy rule and associated conditions.

When multiple "approval" policies apply, the policies are applied in the following order. (The approval policies are on Blueprint provision (approval), on Stack operation (approval), and on Stack resource operation (approval), and on Task remediation):

1. Only the first successful approval policy is applied and no other approval policies are applied.
2. If the applied approval policy has multiple rules, only the first successful rule is used.
3. If a rule has multiple actions, only the first successful action is performed.
4. If the applied approval policy includes both a custom approval and a Service Now approval, only the custom approval process is performed.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform.

- Custom Approval runs the workflow that you specify. The workflow should return a value of approved to complete the operation.
- Service Now Approval supports differing approvers based on the trigger for the policy:

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Available approvers</th>
</tr>
</thead>
<tbody>
<tr>
<td>on Blueprint provision (approval)</td>
<td>• Manager Approval</td>
</tr>
<tr>
<td></td>
<td>• Assignment Group</td>
</tr>
<tr>
<td></td>
<td>• User (lock)</td>
</tr>
<tr>
<td>on Stack operation (approval)</td>
<td>Assignment Group</td>
</tr>
<tr>
<td>on Stack resource operation (approval)</td>
<td>Assignment Group</td>
</tr>
</tbody>
</table>
5. On the Approval form, specify a unique and meaningful **Action Name**.

Example: Action in the rule

6. If you selected **Custom Approval**, specify the workflow and then click **Submit**. If you selected **ServiceNow Approval**, specify who should approve the cloud activity.

**Approval fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager Approval</td>
<td>One or all of the following:</td>
</tr>
<tr>
<td></td>
<td>• Select the Manager Approval check box to require the manager of the approver to also approve the request. The default approval workflow goes to the manager first, then to the group, and finally to the user.</td>
</tr>
<tr>
<td></td>
<td>• Select a user group from the Assignment Group list.</td>
</tr>
<tr>
<td></td>
<td>• Unlock the User lock, select a user from the list, then close the lock.</td>
</tr>
</tbody>
</table>
Create an action for an ‘on Catalog item launch’ policy

The on Catalog item launch trigger fires when an order form (stack request form) is launched for a catalog item. A policy that is triggered by the on Catalog item launch trigger can run a script or override a user-requested value (text values only).

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups.
- Configure a cloud policy rule and associated conditions.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform, enter a unique and meaningful Action Name, and then fill in the form for the action.
Create Policy Rule Action

Execute a Script

This action will execute a ServiceNow script on the instance. All the operation parameters are available as inputs to the script.

Property Override

This action will set the value of a given parameter of the operation.

Settings for the 'Execute a Script' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>Select a category.</td>
</tr>
<tr>
<td>Action Script Name</td>
<td>Specify a unique and meaningful name for the script.</td>
</tr>
<tr>
<td>Action Script</td>
<td>Create the script in the text box. See Create a policy action script for details.</td>
</tr>
</tbody>
</table>

Settings for the 'Property Override' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
<td>Specify the name of the property (attribute) on the user request form to override.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Value</td>
<td>Enter a value that overrides the value in the Property field. You can override text values only. You can specify a static value, an expression, or both. The example action, named SetTheCostCenter, specifies the value Marketing for the CostCenter property.</td>
</tr>
</tbody>
</table>

### Property Override

- **Policy**: ProvisionActions
- **Rule Name**: FirstRuleProvision
- **Action Type**: Property Override
- **Action Name**: SetTheCostCenter
- **Property**: CostCenter
- **Value**: Marketing
- **Is Script Based**: False

**Note:**
When a value is overwritten by both a policy rule and a form rule, the value in the form rule is used.

Expressions can perform the following actions (see Using expressions in Cloud Management for details):

- Set form data values using definition expressions. For example: `${parameter.formData.CatalogAttributeType}
- Assign user data values using definition expressions. For example: `${parameter.userData.userId}
- Set stack or table values using runtime expressions. For example: `${ci.sn_cmp_ip_pool[subnet=${parameter.formData.SubnetId}]}
- Associate a random number with a field using static expressions. Use: `${randomNumber}

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is Script Based</td>
<td>Select the check box to display the <strong>Script</strong> text box and then specify the script. You can use the following example script snippet to override a stack name. The function( formData) section of the script modifies the values for fields on the form. MyStack is the stack name in this example.</td>
</tr>
</tbody>
</table>

```javascript
customScript : function( formData){
  // Manipulation of form parameter is only supported here.
  // Change in any other attributes will be ignored
  // data available for manipulation are
  // Form Data - Ex. StackName can be accessed
  // through formData.StackName
  formData.StackName = formData.StackName = "MyStack";
  // User Data - Ex. User Id can be accessed
  // through this.parameters.userData
  // if(this.parameter.userAgentId == 'servicenowuserId')
  formData.StackName = "MyStack";
  return formData;
},
```

Create an action for an 'on Catalog item request start/end' policy

The **on Catalog item request start** trigger fires after a user opens a request form. The **on Catalog item request end** trigger fires after a user submits a request form. A policy that is triggered by the **on Catalog item request start** or **on Catalog item request end** trigger can run a script or execute a workflow.

- **Role required**: sn_cmp.cloud_governor or admin
- **Optional**: *Configure a cloud policy rule* and associated conditions.

1. In the Cloud Admin Portal, navigate to **Govern > Policies**.
2. Open a cloud policy. Set the policy to the **Draft** state if needed.
3. Open the rule that should perform the action and then click **New** on the Policy Rule Actions related list.
4. On the popup, click **Create** for the type of action to perform, enter a unique and meaningful **Action Name**, and then fill in the form for the action.

### Settings for the 'Execute a Script' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>Select a category.</td>
</tr>
<tr>
<td>Action Script Name</td>
<td>Specify a unique and meaningful name for the script.</td>
</tr>
<tr>
<td>Action Script</td>
<td>Create the script in the text box. See <a href="#">Create a policy action script</a> for details.</td>
</tr>
</tbody>
</table>
Configuring an 'Execute a Workflow' action to run the base-system 'Change Request' workflow

**Settings for the 'Execute a Workflow' action**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Name</td>
<td>Specify a unique and meaningful name for the action that starts the workflow.</td>
</tr>
<tr>
<td>Workflow</td>
<td>Select the workflow to execute.</td>
</tr>
</tbody>
</table>
Create an action for an 'on Lease end' policy

A policy that is triggered by the on Lease end trigger can send a notification or perform a Start, Stop, or Deprovision life cycle operation.

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups.
- Configure a cloud policy rule and associated conditions.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform, enter a unique and meaningful Action Name, and then fill in the form for the action.

### Settings for the 'Run an Operation' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blueprint Operation</td>
<td>Specify Start, Stop, or Deprovision.</td>
</tr>
</tbody>
</table>

### Settings for the 'Send a Notification' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notification</td>
<td>Select the text for the notification:</td>
</tr>
<tr>
<td></td>
<td>- Lease End Date Reached</td>
</tr>
<tr>
<td></td>
<td>- Stack Decommissioned (Lease End)</td>
</tr>
<tr>
<td></td>
<td>- Upcoming Lease End</td>
</tr>
</tbody>
</table>
Create an action for an 'on Resource operation' policy

The on Resource operation trigger fires during the Orchestration process when a user performs a Start, Stop, or Deprovision life cycle operation on a specific resource. A policy that is triggered by the on Resource operation trigger can override a user-requested attribute value, run a script, call a Cloud API, or perform an IP address management operation.

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups.
- Configure a cloud policy rule and associated conditions.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform, enter a unique and meaningful Action Name, and then fill in the form for the action.
### Settings for the 'Property Override' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
<td>Specify the name of the property (attribute) on the user request form to override.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Value</td>
<td>Enter a value that overrides the value in the <strong>Property</strong> field. You can override text values only. You can specify a static value, an expression, or both. The example action, named <strong>SetTheCostCenter</strong>, specifies the value <strong>Marketing</strong> for the <strong>CostCenter</strong> property.</td>
</tr>
</tbody>
</table>

**Property Override**

- **Policy**: ProvisionActions
- **Rule Name**: FirstRuleProvision
- **Action Type**: Property Override
- **Action Name**: SetTheCostCenter
- **Property**: CostCenter
- **Value**: Marketing
- **Is Script Based**: No

**Note:**
When a value is overwritten by both a policy rule and a form rule, the value in the form rule is used.

Expressions can perform the following actions (see [Using expressions in Cloud Management](#) for details):

- Set form data values using definition expressions. For example: `${parameter.formData.CatalogAttributeType}`
- Assign user data values using definition expressions. For example: `${parameter.userData.userId}`
- Set stack or table values using runtime expressions. For example: `$(ci.sn_cmp_ip_pool[subnet=${parameter.formData.SubnetId}])` takes the subnet from the IP Pools table.
- Associate a random number with a field using static expressions. Use: `${randomNumber}`
### Settings for the 'Execute a Script' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>Select a category.</td>
</tr>
<tr>
<td>Action Script Name</td>
<td>Specify a unique and meaningful name for the script.</td>
</tr>
<tr>
<td>Action Script</td>
<td>Create the script in the text box. See <a href="#">Create a policy action script</a> for details.</td>
</tr>
</tbody>
</table>

If you select the **Call Cloud API** action type and save the record, the Policy Rule Action Attributes related list is populated with the attributes to pass to the provider. If you are integrating with Infoblox, you must configure the **DNSSuffix** field. When you create a host, for
example, the value in the **DNSSuffix** field appears in Infoblox for the newly created virtual machine.

### Settings for the ‘Call Cloud API’ action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action Name</strong></td>
<td>Enter a descriptive name for the action. In subsequent policy actions for the same policy, you can reference this name in a <strong>Value</strong> field. It refers to the JSON payload that is received from the cloud provider after an API call. For example, to reference an allocated IP address that was returned by the action that is named <code>RefAction</code>, you can use the following expression in the <strong>Value</strong> field of another policy action: <code>$(parameter.RefAction.Allocated_IP)</code>.</td>
</tr>
<tr>
<td><strong>Provider</strong></td>
<td>Enter the cloud provider from the API library.</td>
</tr>
<tr>
<td><strong>Version</strong></td>
<td>Enter the version from the API library.</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td>Select a CAPI interface. For Infoblox, select <strong>IPAM interface</strong>.</td>
</tr>
<tr>
<td><strong>Operation</strong></td>
<td>Select a CAPI interface operation to perform. For Infoblox, select <code>CreateHostRecord</code> or <code>DeleteHostRecord</code>.</td>
</tr>
<tr>
<td><strong>Credentials</strong></td>
<td>Enter the sys_ID of the necessary credentials. You can also use an expression like: <code>$(ci.sn_cmp_infoblox_server[$(ci.sn_cmp_ip_pool[subnet=${parameter.formData.SubnetId}].provider_instance)].infoblox_server_credential)</code></td>
</tr>
<tr>
<td><strong>Endpoint URL</strong></td>
<td>Enter the endpoint URL of the connector from the API library. For Infoblox, it is the IP address of the Infoblox server.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Settings for the 'IP Address Management' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPAM Method Name</td>
<td>Select the IPAM Method Name. For Infoblox, select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Release IP Address</strong>: Release an IP address from Infoblox when a machine is deprovisioned.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Register IP Address</strong>: Register a new virtual machine to Infoblox. This option is used by AWS and Azure clouds.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Reserve IP Address</strong>: Reserve a new IP address. This option is used by a vSphere cloud.</td>
</tr>
</tbody>
</table>

See [Register IP addresses for AWS and Azure virtual machines in Infoblox](#), [Reserve IP addresses for VMware vSphere virtual machines in InfoBlox](#), and [IPAM integration](#).

---

Create an action for an 'on Resource operation launch' policy

The on Resource operation launch trigger fires before the catalog for a resource operation is loaded from the Cloud User Portal. A policy that is triggered by the on Resource operation launch trigger can run a script or can override a user-requested value (text values only).

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups.
- Configure a cloud policy rule and associated conditions.

1. In the Cloud Admin Portal, navigate to **Govern > Policies**.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform, enter a unique and meaningful **Action Name**, and then fill in the form for the action.
Create Policy Rule Action

Execute a Script

This action will execute a ServiceNow script on the instance. All the operation parameters are available as inputs to the script.

Property Override

This action will set the value of a given parameter of the operation.

Settings for the ‘Execute a Script’ action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>Select a category.</td>
</tr>
<tr>
<td>Action Script Name</td>
<td>Specify a unique and meaningful name for the script.</td>
</tr>
<tr>
<td>Action Script</td>
<td>Create the script in the text box. See Create a policy action script for details.</td>
</tr>
</tbody>
</table>

Settings for the ‘Property Override’ action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
<td>Specify the name of the property (attribute) on the user request form to override.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Value</td>
<td>Enter a value that overrides the value in the Property field. You can override text values only. You can specify a static value, an expression, or both. The example action, named SetTheCostCenter, specifies the value Marketing for the CostCenter property.</td>
</tr>
</tbody>
</table>

**Property Override**

- **Policy**: ProvisionActions
- **Rule Name**: FirstRuleProvision
- **Action Type**: Property Override
- **Action Name**: SetTheCostCenter
- **Property**: CostCenter
- **Value**: Marketing
- **Is Script Based**: no

![Property Override Form](image)

**Note:**
When a value is overwritten by both a policy rule and a form rule, the value in the form rule is used.

Expressions can perform the following actions (see [Using expressions in Cloud Management](#) for details):

- Set form data values using definition expressions. For example: `${parameter.formData.CatalogAttributeType}`
- Assign user data values using definition expressions. For example: `${parameter.userData.userId}`
- Set stack or table values using runtime expressions. For example: `$(ci.sn_cmp_ip_pool[subnet=${parameter.formData.SubnetId}])` takes the subnet from the IP Pools table.
- Associate a random number with a field using static expressions. Use: `${randomNumber}`
Create an action for an ‘on Resource operation request start/end’ policy

A policy that is triggered by the on Resource operation request start or on Resource operation request end trigger can run a script or override a user-requested attribute value.

- **Role required:** sn_cmp.cloud_governor or admin
- **Optional:** Create one or more cloud policy groups.
- **Configure a cloud policy rule** and associated conditions.

- The on Resource operation request start trigger fires after a user submits a resource operation request (Start, Stop, Deprovision).
- The on Resource operation request end trigger fires before completion of a life cycle operation on a resource (Start, Stop, Deprovision).

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the **Draft** state if needed.
3. Open the rule that should perform the action and then click **New** on the Policy Rule Actions related list.
4. On the popup, click **Create** for the type of action to perform, enter a unique and meaningful **Action Name**, and then fill in the form for the action.

---

### Create Policy Rule Action

**Execute a Script**

This action will execute a ServiceNow script on the instance. All the operation parameters are available as inputs to the script.

**Execute a Workflow**

This action will execute a ServiceNow workflow on the instance. The execution will not wait for the workflow to complete.

---

#### Settings for the 'Execute a Script' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>Select a category.</td>
</tr>
<tr>
<td>Action Script Name</td>
<td>Specify a unique and meaningful name for the script.</td>
</tr>
<tr>
<td>Action Script</td>
<td>Create the script in the text box.</td>
</tr>
<tr>
<td></td>
<td>See <a href="#">Create a policy action script</a> for details.</td>
</tr>
</tbody>
</table>

#### Settings for the 'Execute a Workflow' action

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Name</td>
<td>Specify a unique and meaningful name for the action that starts the workflow.</td>
</tr>
<tr>
<td>Workflow</td>
<td>Select the workflow to execute.</td>
</tr>
</tbody>
</table>
Configuring an ‘Execute a Workflow’ action to run the base-system ‘Change Request’ workflow

Create an action for an ‘on Task Remediation’ policy

The on Task Remediation trigger fires when a user resubmits a failed request. A policy that is triggered by the on Task Remediation trigger can start approval workflows.

- Role required: sn_cmp.cloud_governor or admin
- Optional: Create one or more cloud policy groups.
- Configure a cloud policy rule and associated conditions.

1. In the Cloud Admin Portal, navigate to Govern > Policies.
2. Open a cloud policy. Set the policy to the Draft state if needed.
3. Open the rule that should perform the action and then click New on the Policy Rule Actions related list.
4. On the popup, click Create for the type of action to perform.

- **Custom Approval** runs the workflow that you specify. The workflow should return a value of approved to complete the operation.
- **Service Now Approval** supports differing approvers based on the trigger for the policy:
  - Manager Approval
  - Assignment Group
  - User (lock)

5. On the Approval form, specify a unique and meaningful Action Name.
6. If you selected **Custom Approval**, specify the workflow and then click **Submit**. If you selected **ServiceNow Approval**, specify who should approve the cloud activity.

**Approval fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager Approval</td>
<td>One or all of the following:</td>
</tr>
<tr>
<td></td>
<td>- Select the <strong>Manager Approval</strong> check box to require the manager of the</td>
</tr>
<tr>
<td></td>
<td>approver to also approve the request. The default approval workflow goes</td>
</tr>
<tr>
<td></td>
<td>to the manager first, then to the group, and finally to the user.</td>
</tr>
<tr>
<td></td>
<td>- Select a user group from the <strong>Assignment Group</strong> list.</td>
</tr>
<tr>
<td></td>
<td>- Unlock the <strong>User</strong> lock, select a user from the list, then close the</td>
</tr>
<tr>
<td></td>
<td>lock.</td>
</tr>
</tbody>
</table>
Create a policy action script

Use policy action scripts to get, update, or set values in cloud requests.

Role required: sn_cmp.cloud_governor or admin.

For more information on using scripts, see [Cloud scripts and cloud script templates](#).

1. On the Cloud Admin Portal, navigate to **Govern > Policies**.
2. On the **Policy Action Scripts** tab, click **New**, enter a unique and descriptive **Name** and **Description**, fill in the form, and then click **Submit**. After you enter a name, a script appears with a variable that uses the name that you entered.

The following example script determines whether the user belongs to a user group and then overrides the user-specified application server node name value with the value **MyNodeName**.

```javascript
customScript : function(formData){
    var groups = this.parameters.userData.groups;
    if( groups.length > 0){
        formData.App_Server_NodeName = "MyNodeName";
    }
    return formData;
},
```

This example returns a random number:

---

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
This example script adds three hours to the lease date using the base-system `GlideDateTime` class.

```javascript
var SetLeaseShort = Class.create();
SetLeaseShort.prototype = Object.extendsObject(sn_cmp.PolicyExecutionBase, {
  customScript : function(formData){
    // Manipulation of form parameters are supported. Changes in
    // any other attributes
    // will be ignored.
    // The data available for manipulation is as follows:
    // Form Data - for example: StackName can be accessed through
    formData.StackName
    // formData.StackName = "MyStack";
    // User Data - for example: User Id can be accessed through
    this.parameters.userData
    // if(this.parameter.userData.userId == 'servicenowuserId')
    //   this.info("id = " + formData.StackName);

    var leaseValue = now();
    leaseValue = leaseValue.setHours(leaseValue.getHours()+4);
    formData.LeaseEndDate = leaseValue;

    if (formData.hasOwnProperty("LeaseEndDate")) {
      gdt = new GlideDateTime();
      // 10800 = 3 hours
      gdt.addSeconds(10800);
      formData.LeaseEndDate = gdt.getDisplayValue();
    }
    return formData;
  },
  execute: function() {
    if(this.parameters != null && this.parameters.formData != null){
      var inputData = JSON.parse(JSON.stringify(this.parameters.formData));
      var outputFormData = this.customScript(inputData);
      if( outputFormData != null){
        this.parameters.formData = outputFormData;
      }
    }
    var output = {};
    output.answer = this.parameters;
    return JSON.stringify(output);
  },
  type : 'SetLeaseShort'
});
var SetLeaseShortObj = new SetLeaseShort(inputAttributes);
var outputParams = SetLeaseShortObj.execute();
gs.info( "Output of script is = " + outputParams);
```
This example script sets the user-specified Azure Wordpress DNS name field from the form to the same name as the stack name:

```javascript
var WordpressFill = Class.create();
WordpressFill.prototype =
    Object.extendObject(sn_cmp.PolicyExecutionBase, {
        customScript : function(formData){
            // Manipulation of form parameters are supported. Changes in
            // any other attributes
            // will be ignored.
            // The data available for manipulation is as follows:
            // Form Data - for example: StackName can be accessed through
            formData.StackName
            // formData.StackName = "MyStack";
            // User Data - for example: User Id can be accessed through
            this.parameters.userData
            // if(this.parameter.userData.userId == 'servicenowuserId')
            //this.info("id = " + formData.StackName);
            formData.Azure_Wordpress_v6_vmDnsName = formData.StackName;
            return formData;
        }
    });
var WordpressFillObj = new WordpressFill(inputAttributes);
var outputParams = WordpressFillObj.execute();
```

Example policy action script that tags resources
Your instance tracks tagged resources for billing and reporting. Policy action scripts can add and modify resource tags. This example tags a storage volume resource.

Role required: sn_cmp.cloud_governor or admin

See Create a custom tag for cloud resources for information about how to create and use tags in Cloud Management.

2. On the Policy Action Scripts tab, click New and then enter a unique and descriptive Name.
3. Click the lookup icon for Action Script Category, click New to create a new category, and then click Submit.
4. On the Policy Rule Action Script page, enter the script, and then click Submit to save the script.

The script appears in the list.
5. On the Policies tab, click **New**, enter a unique and descriptive **Name** for the policy, fill in the form with the following example settings, and then click **Save**.

<table>
<thead>
<tr>
<th>Policy Trigger</th>
<th>on Resource operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Block</td>
<td>Storage Volume</td>
</tr>
<tr>
<td>Operation</td>
<td>Storage Volume Interface.Provision</td>
</tr>
<tr>
<td>Moment</td>
<td>Pre Operation</td>
</tr>
</tbody>
</table>

6. On the Rules related list, click **New**, enter **Add Custom Tags for Storage** for the **Name**, accept the default settings, and then click **Submit**.

7. On the Policy Rule Actions related list, click **New**.

8. At Execute a Script, click **Create**, fill in the form with the following example settings, and then click **Submit**.

<table>
<thead>
<tr>
<th>Action Name</th>
<th>CreateCustomTagsForStorage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Script Category</td>
<td>ScriptsForCustomTags</td>
</tr>
<tr>
<td>Action Script</td>
<td>AddCustomTagsForStorage</td>
</tr>
</tbody>
</table>

9. Click **Update** on the Policy Rule page.

10. Return to the Policy page and then click **Publish**.

On the Cloud Admin Portal, launch the stack that is associated with the storage resource. You can then confirm that the tags are applied.

Create a cloud approval policy

A cloud approval policy specifies the users who must approve a specified cloud activity before the activity can proceed. Approvers can include the manager of the user making a request, a specified user or group, or users with a specified role. You can specify multiple approvers. Approvals occur in the order that you specify.

- **Role required**: sn_cmp.cloud_governor or admin
- **Optional**: *Create one or more cloud policy groups*.

In this example, a user requests a stack that triggers an approval policy.

1. On the Cloud User Portal, a user submits a request to provision a particular blueprint. The process of provisioning the blueprint is the trigger that causes the policy engine to apply an approval policy. A cloud approval policy specifies the users who must approve a specified cloud activity before the activity can proceed.

2. The policy engine determines that the request meets the condition specified in the rule for the policy. In this example, the condition evaluates to true whenever a particular blueprint is being provisioned.

3. Because the condition is met, the policy engine performs the action that is also specified in the rule. In this example, the action is to create an approval action for the manager.

4. While the approver (the manager) reviews the approval request, the user sees a "waiting for approval" status message on the Cloud User Portal.

5. After the manager approves, the blueprint is provisioned.
The targeted approval policies complement application-wide approval.

- **on Blueprint provision (approval)**: Triggered when the workflow begins to provision a blueprint. The trigger is invoked before the blueprint is provisioned. Because the provisioning process can alter request data (and possibly change costs), however, approval processes run after the blueprint is provisioned.
- **on Stack operation (approval)**: Triggered during any stack operation on the Cloud User Portal.
- **on Stack resource operation (approval)**: Triggered during any resource operation (start, stop, provision, and so on) on the Cloud User Portal.
- **on Task remediation**: Triggered when a user resubmits a failed request.

1. In the Cloud Admin Portal, navigate to **Govern > Policies**.
2. Click **New** and then fill in the form.
Example approval policy

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy name</td>
<td>Enter a descriptive name that includes the word Policy. Do not start the name with a number.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description of the intent of the policy.</td>
</tr>
<tr>
<td>Policy group</td>
<td>Optional: Select a policy group. Each policy in the group is enforced.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Policy trigger | Select a trigger that specifies when the policy should be applied. The following triggers can start approval policies:  
- on Blueprint provision (approval)  
- on Stack operation (approval)  
- on Stack resource operation (approval)  
- on Task remediation |
| Blueprint | Select the blueprint that the policy applies to.  
- If no blueprint is specified, then the policy applies for every blueprint. This setting can decrease performance.  
- You can assign multiple policies to a blueprint. Multiple policies running simultaneously, however, might decrease performance. Publish a policy only when it should be enforced. |
| Operation | Select the operation that the policy applies to. For example, a policy can apply to the Deprovision operation only or to all operations on the blueprint or catalog item.  
**Note:** If no operation is specified, then the policy applies for every operation. This condition can decrease performance.  
- **All**: Any operation executes.  
- **Start**: The resource starts.  
- **Stop**: The resource stops.  
- **Provision**: The resource is provisioned.  
- **Deprovision**: The resource is no longer available to users.  
- **Execute Script**: A script runs on the resource. |
| Start Date / End Date | Specify the start date when the policy should be enforced and the end date when the policy should no longer be enforced. |
| Order of Execution | Specify a number that represents the order in which the policy is applied. A policy with a lower number runs before a policy with a higher number.  
For example, a policy with Order of 100 runs before a policy with an Order of 200. |
| Status | A new policy is in Draft state. Click Publish on the form header to enforce the policy. After a policy is published, you must set it to the Draft state to update it. |

3. Right-click in the header and select Save.
4. When you are ready to enforce the policy, click **Publish**.

**Configure one or more cloud policy rules.**

Create a cloud policy group

A cloud policy group is a container for related policies. Consider grouping policies that are often used together or should be considered together. Grouping policies can help you to apply policies consistently across your organization.

Role required: `sn_cmp.cloud_governor` or `admin`  
1. In the Cloud Admin Portal, navigate to **Govern > Policies**.  
2. Open a cloud policy and click the lookup icon for the **Policy Group**.  
3. Select the group to add the policy to.  
   - To add the policy to an existing group, select the group and then click **Submit**.  
   - To create a policy group to add the policy to, click **New** on the Policy Groups list, specify a unique and meaningful **Name**, and then click **Submit**. The policy is added to the new group.

Export or import a cloud policy

To back up, move, or restore a policy, you can export and import the policy as an update set. The update set includes rules, conditions, actions, scripts, and script categories.

Role required: `sn_cmp.cloud_admin` or `admin`.  
1. On the Cloud Admin Portal, navigate to **Govern > Policies**.  
2. Perform the following action to export a policy:  
   a) Click the export icon for the policy.

The instance generates an update set.

3. To import a policy, install the update set as follows:  
   a) On the instance to import to, navigate to **System Update Sets > Retrieved Update Sets**.
b) Click the **Import Update Set from XML** related link.

c) In the Import XML popup, click **Choose File**, select the file, and then click **Upload**.

If a policy that you exported from the source environment exists in the target environment and both policies have the same `sys_id`, then the policy in the target environment is updated.

**Note:** If the policy that you exported from the source environment exists in the target environment, and the policies have different `sys_ids`, then an error occurs during the import process. You must export a different policy from the source environment or delete the policy with the same name in the target environment.

---

**Pool governance for Cloud Management**

A resource pool is a query or script that filters a table. You configure a resource pool to limit the values that are available to users when they request a catalog item.

**Example use of a pool**

To limit a user to only the networks in a particular logical datacenter, use the resource pool called `NetworkPool` that runs against the Cloud Network (cmdb_ci_network) table. The resource pool uses a script to filter networks based on the LDC that the network belongs to.
How pools work

The naming convention for pools is:

get<thisItem>By<condition>
Where the condition is the top-level entity that forms the relationship between the return values. For example, `getObjectIdByServiceAccount` filters all ObjectIDs that are hosted on the specified service account.

Filters query tables using only the specified condition. Any record that matches the condition is therefore returned.

**Pools and blueprints**

When you configure blueprints, you specify variables for form behavior. You can specify resource pools as the data source for each variable and select which resource pools to use. The variable then uses the filtered values.

**Resource Pool in the base system**

<table>
<thead>
<tr>
<th>Pool</th>
<th>Based on this table</th>
<th>Filter type</th>
<th>Filter name and description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationPool</td>
<td>Application</td>
<td>Query</td>
<td>Filters the names of applications.</td>
</tr>
<tr>
<td>AvailabilityZonePool</td>
<td>Availability zone (cmdb_ci_availability_zone)</td>
<td>Script</td>
<td>Filters availability zones based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• <code>getNameByLDC</code>: Filters availability zones based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• <code>getObjectIdByServiceAccount</code>: Filters availability zones based on the service account they belong to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The Terraform template for IBM requires the name of the availability zone as input for provisioning.</td>
</tr>
<tr>
<td>BusinessServicePool</td>
<td>Service (cmdb_ci_service)</td>
<td>Query</td>
<td>Filters the names of business services.</td>
</tr>
<tr>
<td>ChefServerPool</td>
<td>Chef Server (sn_cfg_chef_server)</td>
<td>Script</td>
<td>Filters chef server credentials by Chef server.</td>
</tr>
<tr>
<td>CloudAccountPool</td>
<td>Cloud Account (cmdb_ci_cmp_cloud_account)</td>
<td>Query</td>
<td>Filters the names of cloud accounts.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><code>getObjectIdByServiceAccount</code>: Filters cloud keypairs based on the service account they belong to.</td>
</tr>
<tr>
<td>CloudKeyPairPool</td>
<td>(cmdb_ci_cloud_key_pair)</td>
<td>Script</td>
<td>Filters cloud keypairs by the logical datacenter and the cloud account it belongs to.</td>
</tr>
<tr>
<td>ComputeProfilePool</td>
<td>Compute Profile (sn_cmp_compute_profile)</td>
<td>Script</td>
<td>Filters compute profiles by the logical datacenter and the cloud account it belongs to.</td>
</tr>
<tr>
<td>CostCenterPool</td>
<td>Cost Center (cmn_cost_center)</td>
<td>Script</td>
<td>Filters to list only the cost centers that the user belongs to.</td>
</tr>
<tr>
<td>DatastorePool</td>
<td>VMware vCenter datastore (cmdb_ci_vcenter_datastore)</td>
<td>Script</td>
<td>Filters datastores based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td>HardwareTypePool</td>
<td>(cmdb_ci_compute_template)</td>
<td>Script</td>
<td>Filters hardware templates based on availability zone they belong to.</td>
</tr>
<tr>
<td>Pool</td>
<td>Based on this table</td>
<td>Filter type</td>
<td>Filter name and description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------</td>
<td>-------------</td>
<td>-------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ImagePool</td>
<td>(cmdb_ci_os_template)</td>
<td>Script</td>
<td>getObjectIdByServiceAccount: Filters the resourceId of OS Image by service account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>In IBM Cloud, OS Images are not specific to a datacenter, so they are hosted at the service account level. Terraform requires the resourceId as input for provisioning.</td>
</tr>
<tr>
<td>IPAddressPool</td>
<td>Cloud IP Address (cmdb_ci_cloud_ip_address)</td>
<td>Query</td>
<td>Filters the IP address</td>
</tr>
<tr>
<td>NetworkPool</td>
<td>Cloud Network (cmdb_ci_network) table</td>
<td>Script</td>
<td>• Filters networks based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• getObjectIdByLDC: Filters network IDs based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td>OSPProfilePool</td>
<td>OS Profile (sn_cmp_os_profile)</td>
<td>Script</td>
<td>Filters compute profiles by the logical datacenter and the cloud account it belongs to.</td>
</tr>
<tr>
<td>ProjectsPool</td>
<td>Project (pm_project)</td>
<td>Query</td>
<td>Filters the names of projects.</td>
</tr>
<tr>
<td>ResourceGroupPool</td>
<td>cmdb_ci_resource_group</td>
<td>Script</td>
<td>Filters resource group based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td>SecurityGroupPool</td>
<td>Compute Security Group (cmdb_ci_compute_security_group)</td>
<td>Script</td>
<td>• getByNetwork: Filters the security group by the network it belongs to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• getObjectIdByServiceAccount: Filters security group IDs based on the service account they belong to.</td>
</tr>
<tr>
<td>SecurityGroupProfilePool</td>
<td>Compute Security Group Profile (sn_cmp_security_grp_profile)</td>
<td>Query</td>
<td>Filters the names of security group profiles.</td>
</tr>
<tr>
<td>SSHKeyPool</td>
<td>CMP SSH Key Pair (sn_cmp_ssh_credentials)</td>
<td>Script</td>
<td>Filters user keys by user.</td>
</tr>
<tr>
<td>StorageAccountPool</td>
<td>Cloud Storage Account (cmdb_ci_cloud_storage_account)</td>
<td>Script</td>
<td>Filters cloud storage accounts based on the logical datacenter they belong to.</td>
</tr>
<tr>
<td>StorageVolumePool</td>
<td>Storage Volume (cmdb_ci_storage_volume)</td>
<td>Query</td>
<td>Filters the names of storage volumes.</td>
</tr>
<tr>
<td>SubnetPool</td>
<td>Cloud Subnet (cmdb_ci_cloud_subnet)</td>
<td>Script</td>
<td>• Filters the subnet by the network it belongs to.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• getObjectIdByNetwork: Filters subnets based on the network they belong to.</td>
</tr>
<tr>
<td>UserGroupPool</td>
<td>Group (sys_user_group)</td>
<td>Script</td>
<td>Returns only the groups that the user belongs to.</td>
</tr>
<tr>
<td>Pool</td>
<td>Based on this table</td>
<td>Filter type</td>
<td>Filter name and description</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------</td>
<td>-------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>VirtualMachinePool</td>
<td>(sn_cmp_resource_pool)</td>
<td>Script</td>
<td><code>getByAvailabilityZone</code>: Filters virtual machine IDs based on the availability zone that they belong to.</td>
</tr>
<tr>
<td>VmFolderPool</td>
<td>VMware vCenter folder (cmdb_ci_vcenter_folder)</td>
<td>Script</td>
<td>Filters VM folders based on the logical datacenter they belong to.</td>
</tr>
</tbody>
</table>

Create a resource pool

Based on blueprint settings, resource pools control the values that a user sees in a catalog item when they request a resource. Only values that pass the pool filter or script appear as options on the catalog item request form.

Role required: `sn_cmp.cloud_governor`

1. In the Cloud Admin Portal, navigate to Manage > Resource Pools.
2. Click New, enter a unique and descriptive Resource Pool Name, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lookup Table</td>
<td>Select the table that contains the records that should appear as options in the catalog item for the resource.</td>
</tr>
<tr>
<td>Lookup Field (Not necessary if using a script. This field is ignored if you configure a script filter.)</td>
<td>Select the table field that contains the values to display to the user. The display value for the lookup field is not used on the Cloud User Portal. Instead, use the Lookup label field(s).</td>
</tr>
<tr>
<td>Lookup label fields (Not necessary if using a script. This field is ignored if you configure a script filter.)</td>
<td>Enter the lookup field label that should appear to the user.</td>
</tr>
</tbody>
</table>

3. Right-click in the header and select Save.
4. On the Resource Pool Filters related list, click New, enter a unique and descriptive Filter Name, and then fill in the form.
The resource pool filter is created.
5. Right-click the form header and select **Save**.

   **Note:** If you are returning filter values from a script in JSON, you do not need to configure filter values. Skip the following step.

6. If you must return a value to the pool and the value is mapped to another table, create a record in the Resource Pool Filter Values related list. On the Resource Pool Filter Values related list, click **New**, fill in the form, and then click **Update**.

   **Note:** If you are returning filter values from a script in JSON, you do not need to configure filter values. Skip this step.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Pool Filter</td>
<td>Enter the name of the filter that the filter value belongs to.</td>
</tr>
<tr>
<td>Type</td>
<td>Enter the type of filter value:</td>
</tr>
<tr>
<td></td>
<td>· Query:</td>
</tr>
<tr>
<td></td>
<td>· Metadata Rule:</td>
</tr>
<tr>
<td>Metadata Rule Name (Appears if you select the Metadata Rule type)</td>
<td>Enter the metadata rule name.</td>
</tr>
<tr>
<td>Operator</td>
<td>Select either:</td>
</tr>
<tr>
<td></td>
<td>· AND</td>
</tr>
<tr>
<td></td>
<td>· OR</td>
</tr>
<tr>
<td>Field</td>
<td>Enter the field that is in the <strong>Lookup Table</strong>. For example, <strong>CloudAccount</strong>.</td>
</tr>
<tr>
<td>Value</td>
<td>Enter the value that you are passing back to the resource pool in the format <strong>$(field)</strong>, where <strong>field</strong> is the value in the <strong>Field</strong> field. For example, <strong>$(CloudAccount)</strong>.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter an order number that determines when the value is applied relative to other resource pool filter values. Lower values are applied first.</td>
</tr>
</tbody>
</table>
The resource pool filter is created.

You can use the resource pool in a blueprint. See Build a Cloud Management blueprint for more information.

**Bind a parameter to an SN pool**

To make catalog ordering less error-prone, you can bind a parameter to an SN pool (a pool provided in the base system). Parameters that are based on an SN pool list only specified values from existing tables on the catalog order form in the Cloud User Portal.

Role required: sn.cmp.cloud_service_designer.

The catalog order form lists only parameters values that have the datasource specified. For example, if you use the Resource Group Name parameter in your template, you can bind the Resource Group Name field to an existing SN pool (ResourceGroupPool) in the metadata section.

When you create a resource block using a cloud template, you can bind a parameter to an SN pool in the metadata section of the Azure Resource Manager (ARM) or CloudFormation (CF) template. All SN pools are listed in the procedure.

1. In the Cloud Admin Portal, navigate to Design > Cloud Templates .
2. Select an existing template or click New to create a new cloud template. If you select an existing template, rename the template.
In the next step, you modify the template definition for the appropriate template type (choose ARM or CloudFormation). You specify the SN pool filter as the **datasource** value.

### SN pools to use as the datasource value

<table>
<thead>
<tr>
<th>SN Pool</th>
<th>Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>AvailabilityZonePool</td>
<td>getAllObjects: Gets all availability zones.</td>
</tr>
<tr>
<td></td>
<td>getObjectsByLDC: Gets availability zone depending on LDC.</td>
</tr>
<tr>
<td>HardwareTypePool</td>
<td></td>
</tr>
<tr>
<td>ImagePool</td>
<td></td>
</tr>
<tr>
<td>LoadBalancerPool</td>
<td>All: Gets all load balancers.</td>
</tr>
<tr>
<td>NetworkPool</td>
<td>getAllObjects: Gets all networks.</td>
</tr>
<tr>
<td></td>
<td>getObjectsByLDC: Gets networks depending on LDC.</td>
</tr>
<tr>
<td>ResourceGroupPool</td>
<td>getObjectsByLDC: Gets resource groups depending on LDC.</td>
</tr>
<tr>
<td>SecurityGroupPool</td>
<td>getObjectsByNetwork: Gets all security groups.</td>
</tr>
<tr>
<td>SecurityGroupProfilePool</td>
<td>All: Gets all security group profiles.</td>
</tr>
<tr>
<td>SSHKeyPool</td>
<td>UserKeys: Gets all SSH keys.</td>
</tr>
<tr>
<td>StorageAccountPool</td>
<td>getObjectsByLDC: Gets storage accounts depending on LDC.</td>
</tr>
<tr>
<td>StorageVolumePool</td>
<td>All: Gets all storage volumes.</td>
</tr>
<tr>
<td>SubnetPool</td>
<td>getObjectsByNetwork: Gets all subnets.</td>
</tr>
<tr>
<td>VirtualMachinePool</td>
<td>getByAvailabilityZone: Gets all virtual machines.</td>
</tr>
</tbody>
</table>

3. Modify an ARM template definition:
   a) Add the attribute **SNC::Parameter::Metadata** in the **parameters** metadata section.
   b) For the **SNC::Parameter::Metadata** attribute, add **datasource** as the key. Enter an SN pool filter for the value. `StorageAccountPool.getObjectsByLDC` in this example.

```json
"parameters": {
  "newStorageAccountName": {
    "type": "string",
    "metadata": {
      "description": "Unique DNS Name for the Storage Account where the Virtual Machine's disks will be placed.",
      "SNC::Parameter::Metadata": {
        "allowedPattern": "[0-9a-z]{1,11}",
        "ConstraintDescription": "must be a alphanumeric",
        "datasource": "ServiceNow::Pools::StorageAccountPool.getObjectsByLDC"
      }
    }
  }
}
```
4. Modify a CloudFormation template definition:
   a) Add the attribute `SNC::Parameter::Metadata` and define the custom attribute name (`VpcId`) along with a `datasource`. For the `datasource` value, enter an SN pool filter. `NetworkPool.getObjectsByLDC` in this example.
   b) Define the `datasourceFilter` mapping attribute to bind the custom name (`VpcId`) to the actual attribute name (`Network`).

   ```json
   "Parameters": {
   "VpcId": {
   "Type": "AWS::EC2::VPC::Id",
   "Description": "VpcId of your existing Virtual Private Cloud (VPC)",
   "ConstraintDescription": "must be the VPC Id of an existing Virtual Private Cloud."
   },
   "SubnetId": {
   "Type": "AWS::EC2::Subnet::Id",
   "Description": "SubnetId of an existing subnet (for the primary network) in your Virtual Private Cloud (VPC)",
   "ConstraintDescription": "must be an existing subnet in the selected Virtual Private Cloud."
   },
   "Metadata": {
   "SNC::Parameter::Metadata": {
   "VpcId": {
   "datasource": "ServiceNow::Pools::NetworkPool.getObjectsByLDC",
   "datasourceFilter": {"Network": "VpcId"}
   },
   "SubnetId": {
   "datasource": "ServiceNow::Pools::SubnetPool.getObjectsByNetwork",
   "datasourceFilter": {"Network": "VpcId"}
   }
   }
   }
   }
   }
   ```

5. Click Submit.
6. After the resource block, blueprint, and catalog items are created, the cloud service user sees only the list of values of the pool data for the parameter in the catalog order form.

Example resource pool that limits choices to cost center
You can use resource pools with blueprints to limit the choices on the cloud catalog request form.

**Use case: Restrict cost center selection**

In this example, the cost of the cloud asset is charged against the budget of the cost center of the user. The base-system `UserCostCenter` resource pool ensures that a user can select only resources in their cost center.
Assumptions

- The Cost Management (con.snc.cost_management) plugin is active.
- Cost centers are defined and users are associated with the cost centers.
- At least one blueprint is defined. This example uses a blueprint named AWS Virtual Server.
- You are assigned the sn_cmp_cloud_admin role and know JavaScript and JSON scripting.

Components

Review resource pool filter

2. Open the CostCenterPool and review the related Resource Pool Filters.
   - All is a query filter that returns all cost centers in the table.
   - UserCostCenter is a script filter that looks up the cost center associated with the user who is ordering the item.

Here is the script for the UserCostCenter filter:

```javascript
getFilteredRecords();
//Do not remove function declaration
/**
 * @returns filtered records in the format
 * [{"value":"lookupValue",label:"displayValue"}]
 */
function getFilteredRecords() {
    var filteredRecords = [];
    var userId = gs.getUserID();
    var userGr = new GlideRecord('sys_user');
    if (userGr.get(userId)){
        var costCenterId = userGr.getValue('cost_center');
        if (costCenterId){
            var costCenterGr = new GlideRecord('cmn_cost_center');
            if (costCenterGr.get(costCenterId)){
                var costCenter = {};
                costCenter.value = costCenterGr.getUniqueValue();
                costCenter.label = costCenterGr.getValue('name');
                filteredRecords.push(costCenter);
            }
        }
    }
    //force to string
    return new global.JSON().encode(filteredRecords);
}
```

Blueprint catalog form parameters

1. Navigate to Design > Blueprints, and then click the tile for the blueprint you want to open.
2. With the blueprint in Draft state, click the Provision operation tile on the Catalog > Request Operation tab.
3. In the Variable sets related list, click the **General Info** variable set. By default, the CostCenter variable is in this variable set.

4. In the Cloud Variables related list on the Variable Set form, click the **CostCenter** variable.

5. On the Cloud Variable form, click the **Type Specifications** tab.

6. Look at the **Pool** and **Pool Filter** fields that refer to the resource pool and filter.
   - **CostCenterPool** is the name of the resource pool.
- **UserCostCenter** is the filter script that pulls in the cost center options for the user to select from.

7. Set the blueprint to **Published**.

**Cost center user**

Identify a user who is a member of a cost center and who has access to the Cloud User Portal.
Testing the resource pool filter

After reviewing the components that comprise this use case, test the cloud catalog item to verify that users can select only their cost center.

1. Impersonate the user, Alene Rabeck in this example.

2. On the Cloud User Portal, click Launch a Stack, and then select the cloud catalog item (AWS Virtual Server in this example).

3. Review the selections in the Cost Center list.
With the `CostCenterPool::UserCostCenter` datasource value for this catalog item, the only option for the Cost Center is the cost center the user is a member of.

**Changing the resource pool filter**

Test that the resource pool filter is controlling the behavior of the Cost Center field by changing it and viewing the results.

1. On the Cloud Admin Portal, navigate to Design > Blueprints and then click AWS Virtual Server.
2. Click the **Provision** operation tile.
3. In the Variable sets related list, click the **General Info** variable set. By default, the CostCenter variable is in this variable set.
4. In the Cloud Variables related list on the Variable Set form, click the **CostCenter** variable.
5. On the Cloud Variable form, click the **Type Specifications** tab.
6. Edit the **Pool filter** field to change the filter from **UserCostCenter** to **All**.

![Type Specifications tab](image)

7. Click **Update**, then click **Publish**.
8. Impersonate the user, **Alene Rabeck** in this example.
9. On the Cloud User Portal, launch a stack, and then select **AWS Virtual Server**.
10. Verify that all cost centers are listed.
Quota governance for Cloud Management
Quotas are limitations or requirements for groups and users on a resource block. Use quotas to prevent wasteful resource usage in your cloud environment.

You apply quotas to a resource, such as a virtual server, datacenter, or a generic cloud resource, and to users of a particular group. You can set a per-user limit, and a total limit for the group for the resource. For example, a quota might limit the number of virtual servers allowed for users in the Database group.

After this kind of quota is set, users who try to provision additional virtual servers will receive an error message and they will not be able to proceed with provisioning virtual servers until the resource count is under the quota limit. If a user belongs to more than one group, they receive the total quota limit from both groups combined.

Note: You can also set per-datacenter limitations on resources by using capacity limits in cloud accounts. See Set capacity limits on user requests for resources.
Resource block types, quota types, and attributes

You can assign quotas to users or groups for the following resource block types:

- Virtual Servers
- vCPUs
- Storage Volume Size
- Network

Quota types include:

**Sum**
The sum of all attributes for a resource, such as the sum of all virtual CPUs.

**Count**
The total number of attributes for a resource, such as the number of virtual machines.

Create a cloud quota definition
A cloud quota definition specifies the limitations on a resource block. You can use templates or manually create different types of configurations for each quota definition.

Role required: sn_cmp.cloud_governor

1. In the Cloud Admin Portal, navigate to **Govern > Quota**.
2. Click **New**, enter a unique and meaningful **Name** and **Description** and then fill in the form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quota Definition Type</td>
<td>Enter a unique identifier for the quota. Typically it is the resource name_count or resource name_size. For example: Network_Count, VM_Count, or StorageVolume_Size.</td>
</tr>
<tr>
<td>Quota Type</td>
<td>Type of quota restriction to set:</td>
</tr>
<tr>
<td></td>
<td>- Count</td>
</tr>
<tr>
<td></td>
<td>- Size</td>
</tr>
<tr>
<td>Attribute</td>
<td>Enter the column name from the underlying CI type on the resource block that holds the value to restrict with the quota. For example, if your quota limits the CPU count on a virtual server, the table of the underlying CI type is cmdb_ci_vm_instance. The cpus column holds that CPU data. So you would enter cpus. You can also enter instance.</td>
</tr>
<tr>
<td>Resource Name</td>
<td>Select the resource block to apply the quota to. See Resource blocks in Cloud Management for more information.</td>
</tr>
<tr>
<td>Template (optional)</td>
<td>Enter the table name of a resource type, for example Hardware Type [cmdb_ci_compute_template] or Image [cmdb_ci_os_template].</td>
</tr>
</tbody>
</table>
3. **Right-click the header and select** **Save.**  
The Quota Configurations and Default Quotas related lists appear.

4. **Click New** in the Quota Configurations related list, enter a unique and meaningful **Group Name**, and then fill in the form:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>Provide a name for the configuration group.</td>
</tr>
<tr>
<td>Group Maximum Limit</td>
<td>Enter a maximum limit for the quota.</td>
</tr>
<tr>
<td>Quota Definition Type</td>
<td>The quota definition name.</td>
</tr>
<tr>
<td>Per User Limit</td>
<td>Enter the maximum limit for each user.</td>
</tr>
</tbody>
</table>

5. **Click Submit.**

6. **You can specify optional default quotas that will be set on the resource block by clicking** **New** **in the Default Quotas related list, and then filling in the form:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Maximum Limit</td>
<td>Enter a maximum limit for the quota.</td>
</tr>
<tr>
<td>Quota Definition</td>
<td>The quota definition for which you are setting the default values.</td>
</tr>
<tr>
<td>Per User Limit</td>
<td>Enter the number of users that this quota applies to.</td>
</tr>
</tbody>
</table>

7. **Click Submit.**

**Budget-based notification and approval**

Administrators can assign a budget for a group and a user within the group. When the user or group reaches the budget limit threshold, notifications are sent alerting them about it.

Administrators can assign a budget (in USD) on a weekly, monthly, quarterly, or yearly basis. The budget set for a user and for a group are independent of each other. For example, a group consisting of five users can have a budget of 100 dollars and each user in that group can be assigned a limit of 25 dollars. An organization decides on the frequency of the budget and all the groups in that organization follow the same frequency. For example, if an organization decides on a monthly budget, then all the groups and users in that organization follow a monthly budget. See **Configure budgets.**

A default budget is given to each new group and new user. A new group gets a default budget of 1000 dollars and a new user gets a default budget of 100 dollars.

Administrators can increase or decrease the budget at any given time. Notifications are sent to the user and the group when the budget limit reaches its threshold limits. Notifications trigger as part of the billing process. Billing discovery is scheduled for each user and group. At the end of
the billing discovery, a comparison is made between the budget limit and the actual cost and if the threshold has reached or has exceeded, notifications are triggered.

Administrators can set up a policy whereby if the budget limit reaches a particular threshold or exceeds the limit, the administrator gives an approval for the user or the group to continue using the resources. For example, the administrator can create a policy for a group whereby when the group's budget threshold reaches 90%, an approval is required for the group to continue consuming resources. If the administrator does not set up a policy for the budget, the user or the group can continue using the resources.

You can view the budget details on the at Analyze > Budget Consumption. You can view the budget consumption details for a user as well as for groups.

Configure budgets
You can configure budgets for groups and users within that group. You can set up a budget period for the group, allocate a budget limit to a group and to each user within that group.

Role required: admin
1. In the Cloud Admin Portal, navigate to Govern > Budget.
2. Click New next to Budget Configuration.
3. Fill in the form fields (as shown in the table)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>Select a group from the list.</td>
</tr>
<tr>
<td>Budget Period</td>
<td>Select a budget period for the group: weekly, monthly, quarterly, or yearly.</td>
</tr>
<tr>
<td>Group Max Limit</td>
<td>Enter a budget limit for the group. By default, the budget limit for each new group is 1000 dollars.</td>
</tr>
<tr>
<td>Per User Limit</td>
<td>Enter a budget limit for each user in that group. By default, the budget limit for each new user is 100 dollars.</td>
</tr>
</tbody>
</table>

4. Click Submit.

Use the Cloud Operations Dashboard
The Cloud Operations Dashboard breaks down cloud service requests from your end users and cloud stacks that you offer in the Cloud User Portal.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin
1. In the Cloud Admin Portal, navigate to Operate > Cloud Operations Dashboard.

The Cloud Operation Dashboard appears, showing you Cloud Service Requests and Stacks. Requests are also broken down by requester in the bar chart below. Stacks are broken down by datacenters.
An example Cloud Operations Dashboard

2. Do any of the following to obtain the tag data you want on the report:

<table>
<thead>
<tr>
<th>Goal</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>See data grouped by another attribute</td>
<td>Select a value in the <strong>Group by</strong> choice list for either chart.</td>
</tr>
<tr>
<td>See updated data</td>
<td>Point to the top of any of the charts until the <strong>refresh icon</strong> (⇘) appears, and then click the icon.</td>
</tr>
<tr>
<td>Save an image of a chart</td>
<td>Point to any of the charts until the options icon (≡) appears, and then select <strong>Save as PNG</strong> or <strong>Save as JPEG</strong>.</td>
</tr>
</tbody>
</table>
Use Cloud Root Cause Analysis reports
Root Cause Analysis reports help you troubleshoot issues with Cloud Management. Use the reports to view details about the Cloud Orchestration Trail and the Cloud API Trail.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to **Operate > Root Cause Analysis Dashboard**.

   The Root Cause Analysis dashboard appears, with the **Cloud Root Cause Analysis Dashboard** displayed by default. This gives you a graphical summary of the Cloud Orchestration Trail with trail records under the reports. These charts group records by different attributes:

   - **Trails**: groups data by the template, which is the component, such as a resource block or blueprint, on which a Cloud Management action was applied.
   - **Stages**: groups data by the stage, which identifies when the transaction took place.
   - **Components**: groups data by the component in the system that was involved in the transaction.
   - **Steps**: groups data by the step, which is the descriptive step associated with a stage and component.

   See [The Cloud Orchestration Trail](#) for a list of possible steps.

   Requests are also broken down by requester in the bar chart below. Stacks are broken down by datacenters.

2. Do any of the following to obtain the tag data you want on the report:
Goal | Do this
---|---
**Filter the data** | Select from the options, such as the **Level**, **Date**, and the **Request Item**.

**See updated data** | Point to the top of any of the charts until the refresh icon ( ) appears, and then click the icon.

**Save an image of a chart** | Point to any of the charts until the options icon ( ) appears, and then select **Save as PNG** or **Save as JPEG**.

3. **Click any Cloud Orchestration API ** | **Number** to view that specific record. See *The Cloud Orchestration Trail* for field descriptions.

4. **To view reports about the Cloud API Trail, click the ** | **Cloud API Dashboard** tab.

   This dashboard shows you several reports for the Cloud API Trail grouped by provider, status, and connectors. You can also see information the methods that were invoked and the API invocations today.
5. Click any the data in the graphs to open the matching Cloud API Trail records. See The Cloud API Trail for descriptions of the Cloud API Trail form fields.

The Cloud Orchestration Trail
The Cloud Orchestration Trail is an activity log for all cloud resource activity on the instance.
### Cloud Orchestration Trail contents

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>An auto-generated number for the trail entry.</td>
</tr>
<tr>
<td>Level</td>
<td>Error</td>
</tr>
<tr>
<td>Template</td>
<td>Azure Datacenter</td>
</tr>
<tr>
<td>Target Instance</td>
<td>East US</td>
</tr>
<tr>
<td>User</td>
<td></td>
</tr>
<tr>
<td>User Group</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Handle Cloud API Respon</td>
</tr>
<tr>
<td>Message</td>
<td>CAPI Failure correlationid mid_user-9bef1b5f-a23d-4324-8062-34de0062473a</td>
</tr>
<tr>
<td>Message Details</td>
<td>&quot;Failed to execute API - Failed with status code and message: 400: {&quot;error&quot;: &quot;code&quot;: &quot;NoRegisteredProviderFound&quot;, &quot;message&quot;: &quot;No registered resource provider 'eastus' and API version '2017-07-29' for type 'storageAccounts'. The supported api-views are 'preview', 2018-02-01, 2017-10-01, 2017-06-01, 2016-12-01, 2016-05-01, 2016-01-01, 2015-11-01, 2015-10-01, 2015-06-01} (script_include:Cloud)&quot;</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Level</td>
<td>One of the following:</td>
</tr>
<tr>
<td></td>
<td>· Debug</td>
</tr>
<tr>
<td></td>
<td>· Info</td>
</tr>
<tr>
<td></td>
<td>· Warn</td>
</tr>
<tr>
<td></td>
<td>· Error</td>
</tr>
<tr>
<td>Template</td>
<td>The component, such as a resource block or blueprint, on which a Cloud Management action was applied.</td>
</tr>
<tr>
<td>Template Instance</td>
<td>The datacenter in which the cloud resource is located.</td>
</tr>
<tr>
<td>User</td>
<td>The user involved in the transaction.</td>
</tr>
<tr>
<td>User group</td>
<td>The user group that the user belongs to.</td>
</tr>
<tr>
<td>Created</td>
<td>The date and time that the system created the entry.</td>
</tr>
<tr>
<td>Milestone</td>
<td>If this transaction is a milestone. You can configure the form to see this field.</td>
</tr>
<tr>
<td>Request Item</td>
<td>The service catalog request generated for the transaction.</td>
</tr>
<tr>
<td>Step</td>
<td>The step that identifies additional information about the transaction, including the component and the stage. You can configure the form to see this field.</td>
</tr>
<tr>
<td>Catalog Item</td>
<td>The catalog item, if any, that was involved in the execution.</td>
</tr>
<tr>
<td>Message</td>
<td>The error message or successful completion message. This could include items like the ID of the corresponding message in the Cloud API Trail record or the Cloud API that was invoked.</td>
</tr>
<tr>
<td>Message details</td>
<td>Details about the error, such as an HTTP error.</td>
</tr>
</tbody>
</table>

An example of an error message detail is as follows:

```
Failed to list loadbalancer Failed : HTTP error code : 403
```

This indicates that the credentials were incorrect, and Discovery could not access the cloud resource. The load balancer interface throws the first error because the load balancer device is the first device that allows access to the cloud resource.

A corresponding error on the MID-Server is captured in the Cloud API Trail.

**Steps, stages, and components**

Each Cloud Orchestration entry has identifying attributes that explain what the triggering transaction was about. *Steps* group these attributes together and identify the *component* in the system that was involved in the transaction and the *stage* at which the transaction took place.

You can configure the Cloud Orchestration Trail form to show the *Step* field if it is not already visible.
<table>
<thead>
<tr>
<th>Step</th>
<th>Component</th>
<th>Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply API Parameter</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Apply Config Parameter</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Apply Endpoint Binding</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Apply Policy</td>
<td>Policy Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Apply Policy Rule Action</td>
<td>Policy Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Call Cloud API</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Call Cloud API Connector</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Check Cloud Match</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check Datacenter Capacity</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check Resource Block Policies</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check User Permissions</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check User Quota</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Complete Cloud Orchestration</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Execute Cat Item Policy</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Execute Form Rules</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Execute Resource Block Operation</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Execute Resource Instance Script</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Get Quota Details</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Get Response</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Handle Cloud API Response</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Handle Policy Execution Response</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Handle Response and Chunking</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Invoke Cloud Orchestration</td>
<td>Workflow Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Invoke Policy Engine</td>
<td>Workflow Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Invoke Pre-Process Workflow</td>
<td>Workflow Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Launch Cat Item</td>
<td>Catalog Item</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Notify Caller</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Prepare BP Execution Plan</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Prepare Runnable API</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Publish API Route Request</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Send Cloud API Response</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Show Form with Filters</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Submit Form</td>
<td>Catalog Item</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Validate Inputs</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
</tbody>
</table>
See [Use Cloud Root Cause Analysis reports](#) for instructions on how to access reports related to the Cloud Orchestration Trail.

Open the Cloud Orchestration Trail

Open the Cloud Orchestration Trail to debug and troubleshoot issues like a failed policy or failed discovery of cloud resources.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to **Operate > Trails**.
2. On the **Cloud Orchestration Trail** tab, filter and sort the list of Orchestration API Trail records as needed. If you are looking for something like a failed Discovery, filter the list so the **Level** column shows only entries with **Error**.
3. Click a link in the **Created** column to open the Orchestration API Trail record.
4. In the CAPI Trail Logs related list, open the log record that displays the information you want. For example, open **route_error** or **error_detail** to debug a failed operation.

*The Cloud API Trail*

The Cloud API Trail is an activity log for all activity that uses the Cloud API and goes through the MID Server.
Cloud API Trail contents

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic route ID</td>
<td>An auto-generated ID number for the entry.</td>
</tr>
<tr>
<td>Mid name</td>
<td>The name of the MID Server through which the Discovery was performed.</td>
</tr>
<tr>
<td>Route status</td>
<td>Whether or not the Discovery operation run by the API was successful. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>• success</td>
</tr>
<tr>
<td></td>
<td>• error</td>
</tr>
<tr>
<td></td>
<td>• executing</td>
</tr>
<tr>
<td>Input parameters</td>
<td>SubscriptionID: 48c3e89-06d2-4f41-bdeb-22603be48a43 Location: westus2</td>
</tr>
<tr>
<td>Interface name</td>
<td>Blob Storage Interface</td>
</tr>
<tr>
<td>Invoked by</td>
<td>CMP</td>
</tr>
<tr>
<td>Method name</td>
<td>GetObject</td>
</tr>
<tr>
<td>Provider name</td>
<td>azure-blob</td>
</tr>
<tr>
<td>Version</td>
<td>1.0</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Input parameters</td>
<td>The input parameter that generated the API trail record. This value is usually the datacenter in which the Discovery was run.</td>
</tr>
<tr>
<td>Interface name</td>
<td></td>
</tr>
<tr>
<td>Invoked by</td>
<td>This value is always CMP when running Discovery.</td>
</tr>
<tr>
<td>Method name</td>
<td>The interface operation from the Cloud API that processed this record.</td>
</tr>
<tr>
<td>Provider name</td>
<td>The cloud provider.</td>
</tr>
<tr>
<td>Version</td>
<td>The version specified in the Cloud API.</td>
</tr>
</tbody>
</table>

**CAPI Trail Logs**

The CAPI Trail Logs related list provides more details about Cloud API trail entry. The following types of log keys are available:

<table>
<thead>
<tr>
<th>Log key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>route_data</td>
<td>Information about the Cloud API calls.</td>
</tr>
<tr>
<td>dynamic_route</td>
<td>Information about the actual route the data took, including URIs.</td>
</tr>
<tr>
<td>route_result</td>
<td>The payload received by the instance, or a description of the result of the data transfer. The payload</td>
</tr>
<tr>
<td>chunk_number</td>
<td>The number of data chunks that the instance received.</td>
</tr>
<tr>
<td>route_status</td>
<td>Whether or not the route connection and payload transfer was successful.</td>
</tr>
<tr>
<td>route_error</td>
<td>The error that occurred. For example, the error Failed to list loadbalancer Failed: HTTP error code : 403 means that your credentials were incorrect and Discovery could not access the cloud resource.</td>
</tr>
<tr>
<td>error_detail</td>
<td>More details about the error, including the Cloud Management API and connector that was used in the attempted Discovery, and the errors that the cloud provider threw.</td>
</tr>
</tbody>
</table>

An example of a route_error is as follows:

```
Failed to list loadbalancer Failed : HTTP error code : 403
```

An example of the error_detail entry for the same error is as follows:

```
com.snc.cmp.connector.cloud.loadbalancer.component.LoadBalancerException: Failed to list loadbalancer Failed : HTTP error code : 403
at com.snc.cmp.connector.cloud.loadbalancer.customizer.impl.AWSLoadBalancerCustomizer.listLoadBalancers(AWSLoadBalancerCustomizer.java:56)
at com.snc.cmp.connector.cloud.loadbalancer.component.LoadBalancerProducer.process(LoadBalancerProducer.java:46)
```
These two entries indicate that the credentials were incorrect, and Discovery could not access the cloud resource. The load balancer interface throws the first error because the load balancer device is the first device that allows access to the cloud resource. The org.apache.camel errors indicated routing errors on the Amazon Web Services side.

The corresponding error on the instance side is captured in the Cloud Orchestration Trail.

Open the Cloud API Trail
Open the Cloud API Trail to debug and troubleshoot issues like a failed policy or failed Discovery of cloud resources.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin
1. In the Cloud Admin Portal, navigate to **Operate** > **Trails**.
2. On the Cloud Api Trail tab, filter and sort the list of Cloud API Trail records as needed. If you are looking for something like a failed Discovery, filter the list so the Route Status column shows only entries with error.
3. Click a link in the Created column to open the Cloud API Trail record.
4. In the CAPI Trail Logs related list, open the log record that displays the information you want. For example, open route_error or error_detail to debug a failed operation.

Cloud Management dashboards and reports
Dashboards enable cloud admins and cloud users to view reports like cloud billing data and cloud tag usage.

View cloud costs on the Billing dashboard
The Billing dashboard provides rich summary information on cloud usage, cost trends, and cost aggregates.

Role required: sn_cmp.cloud_admin
1. In the Cloud Admin Portal, navigate to **Analyze** > **Billing** > **Billing Dashboard**.
   The Billing dashboard shows the Cloud Cost tab by default. This tab displays these two graphs:
   - **Cost Trend**: A line graph showing daily costs over time for the selected Usage Date.
   - **Cost Aggregate**: Total costs for each tag value for the selected Group by tag.
Cloud Cost tab on the Billing dashboard

2. Do any of the following to obtain the tag data you want on the report:

<table>
<thead>
<tr>
<th>Goal</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update the billing period</td>
<td>Select a new option from the Usage Data choice list.</td>
</tr>
<tr>
<td>See data grouped by another tag</td>
<td>Select a value in the Group by choice list under either chart.</td>
</tr>
<tr>
<td>Filter by tag values</td>
<td>Select a tag value from the list.</td>
</tr>
<tr>
<td>Save an image of a chart</td>
<td>Point to either donut chart until the options icon ( ) appears, and then select <strong>Save as PNG</strong> or <strong>Save as JPEG</strong>.</td>
</tr>
</tbody>
</table>
3. To view the areas with the highest level of spending, click the **Top Spends** tab. 
Bar charts showing the total cost for each tag display by default. You can further filter the data by **Service Category**, **Provider**, and **Datacenter**.

4. Click any part of the charts to see cost records for the selected criteria.
The list view of the cost records appear. You can view information such as usage quantity and specific cost per usage date.

![Example cost records](image)

**View tagged resources on the Tag dashboard**

The Tag dashboard shows all tagged resources. Use the Tag dashboard to see specific tag values for a group of resources such as stacks or virtual machines.

Role required: sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to **Analyze > Tag**.

The **Tag Dashboard** tab displays the Tag dashboard where all tagged resources are broken down by tag.

---

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
The Tag dashboard

2. Do any of the following to obtain the tag data you want on the report:

<table>
<thead>
<tr>
<th>Goal</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>See tag values for a specific resource group</td>
<td>Click a section of the <strong>Tagged Resources</strong> donut chart.</td>
</tr>
<tr>
<td>See data grouped by another tag</td>
<td>Select a value in the <strong>Group by</strong> choice list under the <strong>Assigned Tag Values</strong> donut chart.</td>
</tr>
<tr>
<td>See updated data</td>
<td>Point to the top of either chart until the refresh icon ( reloading ) appears, and then click the icon.</td>
</tr>
<tr>
<td>Save an image of a chart</td>
<td>Point to either chart until the options icon ( saving ) appears, and then select <strong>Save as PNG</strong> or <strong>Save as JPEG</strong>.</td>
</tr>
</tbody>
</table>

3. To view a specific tag history record, click any section in the **Assigned Tag Values** chart. A list of tag histories appears that matches the CI class for the selected tag value.
## A tag history record

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created</td>
<td>The date the tag history was created.</td>
</tr>
<tr>
<td>Current</td>
<td>If selected, indicates if the current record is the most current.</td>
</tr>
<tr>
<td>CMDB CI</td>
<td>The actual CI in the CMDB. Click the record icon to see the CI form.</td>
</tr>
<tr>
<td>User group</td>
<td>The group tag associated with this CI.</td>
</tr>
<tr>
<td>User</td>
<td>The user tag associated with this CI.</td>
</tr>
<tr>
<td>Application</td>
<td>The application tag associated with this CI.</td>
</tr>
<tr>
<td>Stack</td>
<td>The stack tag associated with this CI.</td>
</tr>
<tr>
<td>Business service</td>
<td>The business service tag associated with this CI.</td>
</tr>
<tr>
<td>ServiceNow instance</td>
<td>The instance tag associated with this CI.</td>
</tr>
<tr>
<td>Cost center</td>
<td>The cost center tag associated with this CI.</td>
</tr>
<tr>
<td>Time provisioned</td>
<td>The date the machine was provisioned.</td>
</tr>
<tr>
<td>Project</td>
<td>The project tag associated with this CI.</td>
</tr>
<tr>
<td>Custom values</td>
<td>Any custom values tagged to the CI.</td>
</tr>
</tbody>
</table>
Tags for cloud resources
Tags categorize cloud resources to provide richer and more detailed tracking and billing report data.

Create tags to enable Cloud Management to automatically categorize cloud resources after they are provisioned. Each tag associates a specific key:value pair with a requested cloud resource, such as Cost Center=Retail. The Cost Center tag, for example, enables you to group billing data on the Cost Trend report by Cost Center.

After the cloud resource is provisioned, the key:value information in tags become part of the resource's metadata and are saved along with all other parameters about the resource. Because a cloud resource can have multiple tags, you can view cost items from multiple perspectives.

Tag definitions
Tag definitions are records that associate a tag name with a value type. These value types are available:

<table>
<thead>
<tr>
<th>Value type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>A value that user enters or selects from a choice list. This value is stored as a service catalog variable. If a table is associated with a variable tag, the values in the field marked as the Display field are displayed to the user when they request the instance. In most cases, this is the Name field. For example, the names of the business services in the Service table display to the user. If no table is specified in the tag definition, the user is presented with a text field.</td>
</tr>
<tr>
<td>Order field</td>
<td>A value that the user submits, but is not presented as a service catalog variable. The available values are taken from the Orders (sn_cmp_order) table.</td>
</tr>
<tr>
<td>Script</td>
<td>A script that obtains a value from the instance, such as the current time on the instance. The script must obtain the value via the base-system API and return the value to the answer variable. For example, to obtain the current time on the instance, the script would be:</td>
</tr>
</tbody>
</table>

```java
answer = new GlideDateTime().getValue();
```

Default tags
These tags are provided by default:

<table>
<thead>
<tr>
<th>Default Cloud Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
</tr>
<tr>
<td>Application</td>
</tr>
</tbody>
</table>
Cloud administrators can also create custom tags to suit your specific needs. For example, you can create a Location tag can to track the location of the resource.

**AWS tags**

Tags generated for Amazon Web Services (AWS) resources are sent to AWS and are also available from your AWS console.

Create a custom tag for cloud resources

Cloud Administrators can create custom tags in addition to the default tags to categorize cloud resources.

Role required: sn_cmp.cloud_admin

If a resource was provisioned prior to the existence of a particular tag, then that tag must be manually added to the record for that resource.

1. In the Cloud Admin Portal, navigate to **Analyze > Tag > Tag Management > Tag Definitions**.
2. Click **New**.
3. Fill out the form fields (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Name to display on screen for the cloud user.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the key. The name is stored in all tables, along with the value.</td>
</tr>
<tr>
<td>Value type</td>
<td>The <em>type of value</em> for this tag.</td>
</tr>
<tr>
<td>Variable name</td>
<td>The name of the variable. This option appears if you select the Variable value type.</td>
</tr>
<tr>
<td>Table (optional)</td>
<td>The table that provides the choices for the Variable value type field. Select a table from the system. If you do not select a table, cloud users are presented with an empty text field for the variable.</td>
</tr>
<tr>
<td>Order field</td>
<td>The field on the request form. Click the down arrow next to <strong>Click to select</strong>, and then select the field from the Orders [sn_cmp_order] table.</td>
</tr>
<tr>
<td>Active</td>
<td>An indicator that specifies if the tag is in use. Select this option to apply the tag to all new requests.</td>
</tr>
</tbody>
</table>
Troubleshooting tools for Cloud Management

The Cloud Management application provides several tools to help you troubleshoot errors you might encounter during the Discovery of cloud accounts, the provisioning and managing of cloud resources, and the updating of the CMDB.

These troubleshooting tools are available:

Cloud Orchestration Trail

The Cloud Orchestration Trail is an activity log for all cloud resource activity on the instance. Use the Cloud Orchestration Trail to find errors that occur with cloud resources, such as credential errors during Discovery or API execution errors.

See Open the Cloud Orchestration Trail for instructions.

Cloud API Trail

The Cloud API Trail is an activity log for all activity that uses the Cloud API and goes through the MID Server. Use the Cloud API Trail to see API invocations and errors related to the route data, specific API route errors, and Java runtime exceptions.

See Open the Cloud API Trail for instructions.

Root Cause Analysis Dashboard

The Root Cause Analysis Dashboard brings together records from the cloud orchestration trail and cloud API trail and presents them in useful, filterable, lists and charts.

See Use Cloud Root Cause Analysis reports for instructions.

Cloud Orchestrations

Cloud Orchestrations are the orders that your instance processed for each attempted operation on a stack. Use Cloud Orchestration records to view status messages for operations that are run on cloud resources and for details about each API step.

See Open cloud orchestrations for instructions.

The Cloud Orchestration Trail

The Cloud Orchestration Trail is an activity log for all cloud resource activity on the instance.

Cloud Orchestration Trail contents

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>An auto-generated number for the trail entry.</td>
</tr>
<tr>
<td>Level</td>
<td>One of the following:</td>
</tr>
<tr>
<td></td>
<td>• Debug</td>
</tr>
<tr>
<td></td>
<td>• Info</td>
</tr>
<tr>
<td></td>
<td>• Warn</td>
</tr>
<tr>
<td></td>
<td>• Error</td>
</tr>
<tr>
<td>Template</td>
<td>The component, such as a resource block or blueprint, on which a Cloud Management action was applied.</td>
</tr>
<tr>
<td><strong>Field</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Template Instance</td>
<td>The datacenter in which the cloud resource is located.</td>
</tr>
<tr>
<td>User</td>
<td>The user involved in the transaction.</td>
</tr>
<tr>
<td>User group</td>
<td>The user group that the user belongs to.</td>
</tr>
<tr>
<td>Created</td>
<td>The date and time that the system created the entry.</td>
</tr>
<tr>
<td>Milestone</td>
<td>If this transaction is a milestone. You can configure the form to see this field.</td>
</tr>
<tr>
<td>Request Item</td>
<td>The service catalog request generated for the transaction.</td>
</tr>
<tr>
<td>Step</td>
<td>The step that identifies additional information about the transaction, including the component and the stage. You can configure the form to see this field.</td>
</tr>
<tr>
<td>Catalog Item</td>
<td>The catalog item, if any, that was involved in the execution.</td>
</tr>
<tr>
<td>Message</td>
<td>The error message or successful completion message. This could include items like the ID of the corresponding message in the Cloud API Trail record or the Cloud API that was invoked.</td>
</tr>
<tr>
<td>Message details</td>
<td>Details about the error, such as an HTTP error.</td>
</tr>
</tbody>
</table>

An example of an error message detail is as follows:

Failed to list loadbalancer Failed : HTTP error code : 403

This indicate that the credentials were incorrect, and Discovery could not access the cloud resource. The load balancer interface throws the first error because the load balancer device is the first device that allows access to the cloud resource.

A corresponding error on the MID-Server is captured in the Cloud API Trail.

**Steps, stages, and components**

Each Cloud Orchestration entry has identifying attributes that explain what the triggering transaction was about. Steps group these attributes together and identify the component in the system that was involved in the transaction and the stage at which the transaction took place.

You can configure the Cloud Orchestration Trail form to show the Step field if it is not already visible.

<table>
<thead>
<tr>
<th><strong>Step</strong></th>
<th><strong>Component</strong></th>
<th><strong>Stage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply API Parameter</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Apply Config Parameter</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Apply Endpoint Binding</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Apply Policy</td>
<td>Policy Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Apply Policy Rule Action</td>
<td>Policy Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
</tbody>
</table>
### Step Table

<table>
<thead>
<tr>
<th>Step</th>
<th>Component</th>
<th>Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call Cloud API</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Call Cloud API Connector</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Check Cloud Match</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check Datacenter Capacity</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check Resource Block Policies</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check User Permissions</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Check User Quota</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Complete Cloud Orchestration</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Execute Cat Item Policy</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Execute Form Rules</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Execute Resource Block Operation</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Execute Resource Instance Script</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Get Quota Details</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Get Response</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Handle Cloud API Response</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Handle Policy Execution Response</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Handle Response and Chunking</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Invoke Cloud Orchestration</td>
<td>Workflow Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Invoke Policy Engine</td>
<td>Workflow Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Invoke Pre-Process Workflow</td>
<td>Workflow Engine</td>
<td>Workflow and Policy Execution</td>
</tr>
<tr>
<td>Launch Cat Item</td>
<td>Catalog Item</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Notify Caller</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Prepare BP Execution Plan</td>
<td>BP DSL Executor</td>
<td>Cloud Orchestration</td>
</tr>
<tr>
<td>Prepare Runnable API</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Publish API Route Request</td>
<td>CAPI Orchestrator</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Send Cloud API Response</td>
<td>CAPI Probe</td>
<td>Cloud API Orchestration</td>
</tr>
<tr>
<td>Show Form with Filters</td>
<td>Dynamic Forms</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Submit Form</td>
<td>Catalog Item</td>
<td>Catalog Item Request</td>
</tr>
<tr>
<td>Validate Inputs</td>
<td>BP DSL Validator</td>
<td>Cloud Orchestration</td>
</tr>
</tbody>
</table>

See #unique_1199 for instructions on how to access reports related to the Cloud Orchestration Trail.

Open the Cloud Orchestration Trail

Open the Cloud Orchestration Trail to debug and troubleshoot issues like a failed policy or failed Discovery of cloud resources.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to **Operate > Trails**.
2. On the **Cloud Orchestration Trail** tab, filter and sort the list of Orchestration API Trail records as needed. If you are looking for something like a failed Discovery, filter the list so the **Level** column shows only entries with **Error**.

3. Click a link in the **Created** column to open the Orchestration API Trail record.

4. In the CAPI Trail Logs related list, open the log record that displays the information you want. For example, open `route_error` or `error_detail` to debug a failed operation.

**The Cloud API Trail**
The Cloud API Trail is an activity log for all activity that uses the Cloud API and goes through the MID Server.

### Cloud API Trail contents

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic route ID</td>
<td>An auto-generated ID number for the entry.</td>
</tr>
<tr>
<td>Mid name</td>
<td>The name of the MID Server through which the Discovery was performed.</td>
</tr>
<tr>
<td>Route status</td>
<td>Whether or not the Discovery operation run by the API was successful. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>- success</td>
</tr>
<tr>
<td></td>
<td>- error</td>
</tr>
<tr>
<td></td>
<td>- executing</td>
</tr>
<tr>
<td>Input parameters</td>
<td>The input parameter that generated the API trail record. This value is usually the datacenter in which the Discovery was run.</td>
</tr>
<tr>
<td>Interface name</td>
<td></td>
</tr>
<tr>
<td>Invoked by</td>
<td>This value is always <strong>CMP</strong> when running Discovery.</td>
</tr>
<tr>
<td>Method name</td>
<td>The interface operation from the Cloud API that processed this record.</td>
</tr>
<tr>
<td>Provider name</td>
<td>The cloud provider.</td>
</tr>
<tr>
<td>Version</td>
<td>The version specified in the Cloud API.</td>
</tr>
</tbody>
</table>

**CAPI Trail Logs**
The CAPI Trail Logs related list provides more details about Cloud API trail entry. The following types of log keys are available:

<table>
<thead>
<tr>
<th>Log key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>route_data</td>
<td>Information about the Cloud API calls.</td>
</tr>
<tr>
<td>dynamic_route</td>
<td>Information about the actual route the data took, including URIs.</td>
</tr>
<tr>
<td>route_result</td>
<td>The payload received by the instance, or a description of the result of the data transfer. The payload</td>
</tr>
<tr>
<td>Log key</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>chunk_number</td>
<td>The number of data chunks that the instance received.</td>
</tr>
<tr>
<td>route_status</td>
<td>Whether or not the route connection and payload transfer was successful.</td>
</tr>
<tr>
<td>route_error</td>
<td>The error that occurred. For example, the error Failed to list loadbalancer Failed : HTTP error code : 403 means that your credentials were incorrect and Discovery could not access the cloud resource.</td>
</tr>
<tr>
<td>error_detail</td>
<td>More details about the error, including the Cloud Management API and connector that was used in the attempted Discovery, and the errors that the cloud provider threw.</td>
</tr>
</tbody>
</table>

An example of a route_error is as follows:

```
Failed to list loadbalancer Failed : HTTP error code : 403
```

An example of the error_detail entry for the same error is as follows:

```
com.snc.cmp.connector.cloud.loadbalancer.component.LoadBalancerException: Failed to list loadbalancer Failed : HTTP error code : 403
at com.snc.cmp.connector.cloud.loadbalancer.customizer.impl.AWSLoadBalancerCustomizer.list
at com.snc.cmp.connector.cloud.loadbalancer.component.LoadBalancerProducer.process
at org.apache.camel.util.AsyncProcessorConverterHelper$ProcessorToAsyncProcessorBridge.process
at org.apache.camel.processor.SendProcessor.process
```

These two entries indicate that the credentials were incorrect, and Discovery could not access the cloud resource. The load balancer interface throws the first error because the load balancer device is the first device that allows access to the cloud resource. The `org.apache.camel` errors indicated routing errors on the Amazon Web Services side.

The corresponding error on the instance side is captured in the Cloud Orchestration Trail.

Open the Cloud API Trail

Open the Cloud API Trail to debug and troubleshoot issues like a failed policy or failed Discovery of cloud resources.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to Operate > Trails.
2. On the Cloud Api Trail tab, filter and sort the list of Cloud API Trail records as needed. If you are looking for something like a failed Discovery, filter the list so the Route Status column shows only entries with error.
3. Click a link in the Created column to open the Cloud API Trail record.
4. In the CAPI Trail Logs related list, open the log record that displays the information you want. For example, open route_error or error_detail to debug a failed operation.

Use Cloud Root Cause Analysis reports

Root Cause Analysis reports help you troubleshoot issues with Cloud Management. Use the reports to view details about the Cloud Orchestration Trail and the Cloud API Trail.
Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin

1. In the Cloud Admin Portal, navigate to **Operate > Root Cause Analysis Dashboard**.

   The Root Cause Analysis dashboard appears, with the **Cloud Root Cause Analysis Dashboard** displayed by default. This gives you a graphical summary of the Cloud Orchestration Trail with trail records under the reports. These charts group records by different attributes:

   - **Trails**: groups data by the template, which is the component, such as a resource block or blueprint, on which a Cloud Management action was applied.
   - **Stages**: groups data by the stage, which identifies when the transaction took place.
   - **Components**: groups data by the component in the system that was involved in the transaction.
   - **Steps**: groups data by the step, which is the descriptive step associated with a stage and component.

   See [The Cloud Orchestration Trail](#) for a list of possible steps.

   Requests are also broken down by requester in the bar chart below. Stacks are broken down by datacenters.

2. Do any of the following to obtain the tag data you want on the report:

<table>
<thead>
<tr>
<th>Goal</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter the data</td>
<td>Select from the options, such as the Level, Date, and the Request Item.</td>
</tr>
</tbody>
</table>
### Goal | Do this
---|---
**See updated data** | Point to the top of any of the charts until the refresh icon (⟳) appears, and then click the icon.

**Save an image of a chart** | Point to any of the charts until the options icon (⋮) appears, and then select **Save as PNG** or **Save as JPEG**.

---

3. Click any Cloud Orchestration API **Number** to view that specific record. See [The Cloud Orchestration Trail](#) for field descriptions.

4. To view reports about the Cloud API Trail, click the **Cloud API Dashboard** tab. This dashboard shows you several reports for the Cloud API Trail grouped by provider, status, and connectors. You can also see information the methods that were invoked and the API invocations today.
5. Click any the data in the graphs to open the matching Cloud API Trail records. See *The Cloud API Trail* for descriptions of the Cloud API Trail form fields.

*Open cloud orchestrations*
Cloud orchestration records show you the orders that your instance processed for each attempted operation on a stack. They also show you the values of the fields that the user submitted through the Cloud User Portal. Use cloud orchestrations to troubleshoot issues that occur when a user provisions a cloud resource or runs another operation on an existing cloud resource.

Role required: sn_cmp.cloud_operator or sn_cmp.cloud_admin

1. On the instance (not the Cloud User Portal), enter the following text into the application filter:
The list of Cloud Orchestrations appear.

2. Sort and filter the list to find the error or other message you are looking for.

   For example, to sort the list by date, click the **Order Date** column. To show only errors that occurred, right-click **Error** in a cell and select **Show Matching**.

3. Open a cloud orchestration record by clicking the **Order Date** field.

4. Review the form fields:
The example form shows you the type of message you can see when an operation fails due to the incorrect credentials. The OrderForm Data field shows the Sys ID of the credential record and the service account ID.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Request Item</td>
<td>A service catalog request, if any, associated with this transaction.</td>
</tr>
<tr>
<td>Context Key</td>
<td>A key identifier for this message.</td>
</tr>
<tr>
<td>Context Object</td>
<td>This field is typically empty.</td>
</tr>
<tr>
<td>Context Instance</td>
<td>The blueprint used in transactions on a stack or the resource block used in transactions on a cloud resource.</td>
</tr>
<tr>
<td>Owner Instance</td>
<td>The Owner Instance is the resource block on which the transaction took place. This field is only populated when the Entity Type is Resource.</td>
</tr>
<tr>
<td>Owner Table</td>
<td>The owner table is always Stack (sn_cmp_stack) for the provisioning of a stack. Otherwise, the table is the CI type of the resource, such as Virtual Machine (sn_cmp_vm_instance).</td>
</tr>
<tr>
<td>Entity Type</td>
<td>Whether the transaction occurred on Blueprint or a Resource (resource block).</td>
</tr>
<tr>
<td>Operation Name</td>
<td>The interface and operation that is triggered.</td>
</tr>
<tr>
<td>Order Date</td>
<td>The date of the transaction.</td>
</tr>
<tr>
<td>OrderForm Data</td>
<td>The data submitted through the user form.</td>
</tr>
<tr>
<td>Tag values</td>
<td>Any tag values involved in the transaction.</td>
</tr>
<tr>
<td>Resource</td>
<td>The resource block on which the operation took place. This field is populated only when the value of the Entity type field is Resource.</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the message. If you are viewing an error, the Error option is listed.</td>
</tr>
<tr>
<td>Stack</td>
<td>The stack that the resource belongs to.</td>
</tr>
<tr>
<td>Status Message</td>
<td>The status message that explains the issue. Search the cloud provider documentation for the status codes and messages that appear in this field. The cloud provider might provide solutions available to you.</td>
</tr>
<tr>
<td>User</td>
<td>The user who triggered the operation.</td>
</tr>
<tr>
<td>Mid name</td>
<td>This field can show the name of the MID Server involved in the transaction. It is often left blank.</td>
</tr>
</tbody>
</table>

**Scoped applications for Cloud Management**

Create scoped applications using the standard Cloud Management entities to support new cloud providers and new configuration management providers. For example, you can create a scoped application for a third-party platform that provisions a virtual machine, such as the Google Compute Engine service.

Use the ServiceNow®Studio to create scoped applications on a regular cadence and publish them to a repository or the ServiceNow Store. Studio specifies plugin dependencies, specifies licensing requirements, and publishes scoped applications to a repository, to any instance, or to the ServiceNow Store. See the ServiceNow Store documentation for details on publishing an application.
Content entity types that are supported for creating scoped applications

- CMDB CI classes
- CMDB CI identifiers
- CMDB relations
- CMDB CI identification entries
- Custom tables
- Cloud APIs
- Resource blocks
- Policies
- Blueprints
- Script includes
- Pools and filters
- MID Server script files
- MID Server capabilities
- Credentials
- Provider credentials mapping
- Provider types (Puppet configuration management provider only)

Create a Puppet Enterprise scoped application for Cloud Management

Create a Puppet Enterprise scoped application for Cloud Management by using a virtual machine (for example, a Linux virtual machine). Creating a scoped application and publishing the application to the ServiceNow Store makes finding the application more efficient.

- See the ServiceNow Store documentation for prerequisites for publishing.
- Create a repository in a version control system like the GitHub hosting service.
- Test the connection to make sure that the Puppet Enterprise server can communicate with the Puppet Enterprise agents.
- Provision a virtual machine (for example, Linux virtual machine) from a blueprint.
- Verify that the Puppet Enterprise configuration management provider is up and running.
- Role required: sn_cmp.cmp_root_admin

1. Create an application in the ServiceNow Studio.
2. Switch to the newly created application to set dependencies and create application settings for the new application.
3. Create a table for storing credential information to access the Puppet Enterprise server.
4. To manage all your Puppet Enterprise instances, create a configuration item (CI) class.
5. Create a cloud API provider.
6. Create cloud APIs.
7. Create script includes to be used by the cloud APIs to communicate with the Puppet Enterprise server.
8. Create a CI class for an environment.
    An environment can be a production, development, or a test environment.
9. Add identification rules to the CI class.
10. Configure metadata rules to create a relationship between the Puppet Enterprise CI class and the Puppet Enterprise environment CI class.
11. Create a resource block.
12. Configure cloud script templates to install a Puppet Enterprise agent on a virtual machine once the virtual machine is provisioned.
13. Create a workload provider type.
14. Create a capability for the Puppet Enterprise provider in the MID Server.
A capability is required for each MID Server to work with the Cloud Management application.

15. Test the scoped application.
16. Publish the scoped application to the ServiceNow Store.

Your Puppet Enterprise scoped application for Cloud Management is created and published to the ServiceNow Store.

Create an application for Puppet provider in the ServiceNow Studio

Create an application for Puppet provider in the ServiceNow Studio. Create all your Puppet provider configuration information in the form of an application using the ServiceNow Studio. Once the configuration information is in an application, you can use the files in your application, create new files, and modify application files in a tabbed environment.

Role required: sn_cmp.cmp_root_admin

1. Navigate to System Applications > Studio.
   The Load Application pop-up window appears.
2. In the Load Application pop-up window, click Create Application.
   The Create Application dialog box appears.
3. In the Create custom application section, click Create.
4. On the form, fill in the fields.

Create Application form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the application.</td>
</tr>
<tr>
<td>Scope</td>
<td>Scope of the application. The scope appears automatically.</td>
</tr>
<tr>
<td>Menu</td>
<td>Menu file name. By default, the menu file name is the name of the application and appears automatically.</td>
</tr>
<tr>
<td>User Role</td>
<td>Required user role. By default, the scope determines the user role and appears automatically.</td>
</tr>
</tbody>
</table>

5. To create a table associated with the application:
   a) Enable the Create Table feature.
   b) On the form, fill in the fields.

Table form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Name of the table.</td>
</tr>
<tr>
<td>Name</td>
<td>File name of the table.</td>
</tr>
<tr>
<td>Module</td>
<td>Module associated with the table.</td>
</tr>
<tr>
<td>Extends Table</td>
<td>Table to be extended.</td>
</tr>
<tr>
<td>Extensible</td>
<td>Indication of whether the table can be extended.</td>
</tr>
<tr>
<td>Live Feed</td>
<td>Indication of whether live feed is in use.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Auto-number</td>
<td>Indication of whether the table is automatically numbered. If selected, prompts for prefix, number, and number of digits.</td>
</tr>
</tbody>
</table>

6. Click **Create**. The Confirm Application dialog box appears.

7. In the Confirm Application dialog box, click **OK**. A successful completion message appears. After the application is successfully created, the Application Creation window opens with the message 'Puppet' service has been created.

Your Puppet provider configuration information is in the form of an application.

**Set dependencies on the Puppet scoped application**

Set dependencies on the Puppet scoped application

Set dependencies and create application settings on the newly created puppet scoped application. Setting dependencies and creating application settings lets you customize the application and helps you manage the application.

Role required: sn_cmp.cmp_root_admin

1. Navigate to **System Applications > Studio** and click **Create Application**. The Load Application window appears.
2. Select the application that you created (for example, **Config Management Puppet**). The Welcome to Studio home page appears.
3. Navigate to **File > Settings**. The Application Settings record appears.
4. From the **Runtime Access Tracking** list, select **None**.
5. Click **Update**.
6. Click the Dependencies related list.
7. Click **Edit**.
8. Associate the plugins with the Config Management Puppet application by moving the desired plugins to the **Dependencies** list.
9. Click **Save**.

All the plugins, added as dependencies, appear in the Dependencies related list.

**Create a table for storing Puppet credential information**

Create a table for storing Puppet credential information

Create a table for storing credential information pertaining to the Puppet provider. The credential information is used to connect to the Puppet server. Having this information in a table helps you manage the information about the connection.

Role required: sn_cmp.cmp_root_admin

1. Navigate to **System Definition > Tables**.
2. Click **New**.
3. On the form, fill in the fields.
### Table form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Label for the table (for example, Puppet Enterprise Credentials). The label appears in the list and form views for the table.</td>
</tr>
<tr>
<td>Name</td>
<td>Table file name (for example, x_sn_config_puppet_puppet_enterprise_credentials). This name is automatically populated based on the table label and a prefix. For a table in a scoped application, the name is prefixed with a namespace identifier to indicate that it is part of an application.</td>
</tr>
</tbody>
</table>
| Extends table      | Table to extend (for example, Credentials). Extending a base table incorporates all the fields of the original table and creates system fields for the new table. You can extend tables that are marked as extensible under one of the following conditions:  
- Both tables are in the same scope.  
- Both tables enable configuration from other scopes. |
| Application        | Applications associated with the table. Global is the default. |
| Create module      | Option for creating a module. This check box is selected by default. |
| Create mobile module | Option for creating a mobile module. This check box is selected by default. If you clear the Create module check box, this check box disappears. |
| Add module to menu | Menu for the new module. If you clear the Create module check box, this check box disappears. |
| New menu name      | Name for the new menu. If you clear the Create module check box, this check box disappears. |

4. Change the names of the table columns.  
   a) Click the Columns related list. The Table Columns form appears.  
   b) On the form, fill in the fields.  

**Note:** You may need to modify the Table Columns form to see the heading names. To modify the form, click the personalize list icon ( ). Move the desired heading names to the Selected list and select the check box options as needed, then click OK.
Table Columns form

<table>
<thead>
<tr>
<th>Column Label</th>
<th>Column Name</th>
<th>Type</th>
<th>Max Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter the value,</td>
<td>Enter the value,</td>
<td>Select string</td>
<td>Enter the value, 100.</td>
</tr>
<tr>
<td>Machine Username</td>
<td>machine_username</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enter the value,</td>
<td>Enter the value,</td>
<td>Select Password (2</td>
<td>Enter the value, 255.</td>
</tr>
<tr>
<td>Machine Password</td>
<td>machine_password</td>
<td>way Encrypted)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. Right-click anywhere in the Table form title bar and click Save.
6. Modify the user privileges.
   a) Click your user name or role in the banner and select Elevate Roles.
   b) Select the security_admin check box and click OK.
   The Tables form appears.
   c) Open the Puppet Enterprise Credentials table.
   d) On the Puppet Enterprise Credentials table record, click the Access Controls related list.
   The list of credential records appears.
   e) Open the record listed with the read operation.
   f) In the Required role section, add a role with the value, mid_server, to the Access Control record.
   g) Click Save and then click Update.

A table containing the Puppet credentials information appears in the Tables list view using the column headings you specified.

Create a CI class for the Puppet provider

Create a CI class for your Puppet provider instances to make managing the provider instances more efficient.

Role required: sn_cmp.cmp_root_admin

1. Navigate to Configuration > CI Class Manager.
2. To display the CI classes list, click Hierarchy.
4. To create a new child class, click **Add Child Class**. The Create Class window appears in the Provide Basic Info stage.
5. In the **Display name** field, enter a name (for example, **Puppet Enterprise**) for the class.

6. To advance to the Add Attributes stage, click **Next**.
   The Create Class window appears in the Add Attributes stage.

7. In the Add Attributes stage, click **Next**.
   The Create Class window appears in the Set Identification Rule stage.
Set Identification Rule stage

8. To create an identification rule, click **Add**. The Create Identification Rule dialog box appears.
9. In the Create Identification Rule dialog box, accept the default values and click **Save**.

10. Click **Next** through each remaining stage until you return to the Config Management Workload Provider form.

The class and the identification rule have been created and are available for you to use.

**Create a cloud API provider for Puppet**

Create a cloud API provider for Puppet. The cloud API provider improves the interaction between Cloud Management and the Puppet provider.

Role required: **sn_cmp.cmp_root_admin**

1. Navigate to **Cloud Management > Cloud Management Portal > Design > Cloud API**. The Cloud API window appears.

2. Click the **Providers** tab if it is not already selected.

3. Add a provider.
   a) In the Cloud API window, click **New**.
   b) On the form, fill in the fields.
Add Cloud Provider form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Provider</td>
<td>Name of the cloud provider.</td>
</tr>
<tr>
<td>Description</td>
<td>Short description for the Puppet cloud provider.</td>
</tr>
<tr>
<td>Datacenter class</td>
<td>Datacenter class selected for the Puppet cloud provider.</td>
</tr>
</tbody>
</table>

c) Click **Submit**.
The Cloud Provider record displays details about the provider.

4. Add a new cloud product.
   a) In the Cloud Products related list, click **New**.
   b) On the form, fill in the fields.

Cloud Product form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the cloud product.</td>
</tr>
<tr>
<td>Service Category</td>
<td>Service category selected.</td>
</tr>
</tbody>
</table>

c) Click **Submit**.

The new cloud product appears in the Cloud Provider record.

Create a cloud API for Puppet

Create a cloud API for Puppet
Create a cloud API for the Puppet configuration management provider. The cloud API helps integrate Cloud Management services with the Puppet provider.

Role required: sn_cmp.cmp_root_admin

1. In the Cloud Admin Portal, navigate to **Design > Cloud API**.
   To access the Cloud Admin Portal, navigate to **Cloud Management > Cloud Admin Portal**.
2. Click **API** and then click **New**.
3. On the form, fill in the fields.

Add Cloud API form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud API</td>
<td>Name of the Cloud API (for example, Puppet Enterprise Configuration API).</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface selected for the cloud API (for example, Configuration Management Interface).</td>
</tr>
<tr>
<td>Connector</td>
<td>Name of the connector. This field is automatically set by default.</td>
</tr>
<tr>
<td>Product</td>
<td>Product for the Cloud API.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Version</td>
<td>Version of the cloud API (for example, 1.0).</td>
</tr>
<tr>
<td>Scripted</td>
<td>Option used to indicate if the API is scripted. This check box is selected by default.</td>
</tr>
<tr>
<td>Script Type</td>
<td>Type of script used by the Cloud API.</td>
</tr>
<tr>
<td>ExecuteScriptOn instance</td>
<td>Option used to indicate whether to execute the script on the instance.</td>
</tr>
</tbody>
</table>

4. Click **Submit**. The record for the new Cloud API appears.

5. Click the **API Config Overrides** related link and then click **New**. The API Config Overrides form appears.

6. Create three API overrides; one for credentials, one for endpoint, and one for identity. Follow the same procedure for each one.
   a) To create the API overrides for credentials, fill in the form.

   **API Config Override for Credential form**

<table>
<thead>
<tr>
<th>Config parameter</th>
<th>Override value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter <strong>Credentials</strong> and then click the lookup list icon (🔍) and select the Config parameter name <strong>Credentials</strong> that corresponds to the Endpoint, <strong>cfg-mgmt</strong>.</td>
<td>$(CloudCredential.password)</td>
</tr>
</tbody>
</table>

   b) Click **Submit**.

   c) To create the API overrides for endpoints, fill in the form.

   **API Config Override for Endpoint form**

<table>
<thead>
<tr>
<th>Config parameter</th>
<th>Override value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter <strong>Endpoint</strong> and then click the lookup list icon (🔍) and select the Config parameter name <strong>Endpoint</strong> that corresponds to the Endpoint, <strong>cfg-mgmt</strong>.</td>
<td>$(CloudCredential.URL)</td>
</tr>
</tbody>
</table>

   d) Click **Submit**.

   e) To create the API overrides for identity, fill in the form.
API Config Override for Identity form

<table>
<thead>
<tr>
<th>Config parameter</th>
<th>Override value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter <strong>Identity</strong> and then click the lookup list icon (🔍) and select the Config parameter name <strong>Identity</strong> that corresponds to the Endpoint, <strong>cfg-mgmt</strong>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$(CloudCredential.user_name)</td>
</tr>
</tbody>
</table>

f) Click **Submit**.

The three API config overrides, credentials, endpoint, and identity are created. Enter scripts for the Cloud API Interface operations: Discover, BootstrapNode, DeregisterNode, ExecuteConfigPackages, and RegisterNode.

**Note:** For the content you can use for the API interface operations, see [Sample scripts for Cloud API interface operations](#).

7. Click the CAPI Method Mappers related list and then click **Discover**.
   a) In the **Request Script** field, click the information icon (🔍) to open the record.
   b) Enter the script, **puppet-enterprise-1.0-Discover**, in the **Script** field and click **Update**.

8. In the CAPI Method Mappers related list, click **BootstrapNode**.
   a) In the **Request Script** field, click the information icon (🔍) to open the record.
   b) Enter the script, **puppet-enterprise-1.0-BootstrapNode**, in the **Script** field and click **Update**.

9. In the CAPI Method Mappers related list, click **DeregisterNode**.
   a) In the **Request Script** field, click the information icon (🔍) to open the record.
   b) Enter the script, **puppet-enterprise-1.0-DeregisterNode**, in the **Script** field and click **Update**.

10. In the CAPI Method Mappers related list, click **ExecuteConfigPackages**.
   a) In the **Request Script** field, click the information icon (🔍) to open the record.
   b) Enter the script, **puppet-enterprise-1.0-ExecuteConfigPackage**, in the **Script** field and click **Update**.

11. In the CAPI Method Mappers related list, click **RegisterNode**.
   a) In the **Request Script** field, click the information icon (🔍) to open the record.
   b) Enter the script, **puppet-enterprise-1.0-RegisterNode**, in the **Script** field and click **Update**.

A cloud API for puppet is created.

Create script includes for a Puppet scoped application
Sample scripts for Cloud API interface operations

By default, sample scripts are provided by the Cloud Management application for the cloud API interface operations: Discover, BootstrapNode, DeregisterNode, ExecuteConfigPackages, and RegisterNode.

**puppet-enterprise-1.0-Discover**

For the Discover cloud API, use the following script:

```javascript
discoverPuppetEnterprise();

function discoverPuppetEnterprise() {
    try {
        var discoveryPayload = {};
        var additionalParameters = this.parameters.get('AdditionalParameters');
        var additionalParams = new JSON().decode(additionalParameters);
        var discoveryParameter = additionalParams.entity;
        ms.log("$$ discoveryParameter : " + discoveryParameter);

        if (discoveryParameter == "environment") {
            // invoke Environments Discovery
            var peE = new PuppetEnterpriseEnvironments(this.parameters, this.headers);
            var environments = peE.discoverEnvironments();
            discoveryPayload['environments'] = environments.environments;
            discoveryPayload['objectId'] = environments.objectId;
        }

        if (discoveryParameter == "group") {
            // invoke Environments Discovery
            var peG = new PuppetEnterpriseClassification(this.parameters, this.headers);
            var groups = peG.discoverGroups();
            discoveryPayload['groups'] = groups.groups;
            discoveryPayload['objectId'] = groups.objectId;
        }

        if (discoveryParameter == "inventory") {
            // invoke Inventories Discovery
            var peI = new PuppetEnterpriseInventory(this.parameters, this.headers);
            var inventories = peI.discoverInventories();
            discoveryPayload['inventories'] = inventories.inventories;
            discoveryPayload['facts'] = inventories.facts;
            discoveryPayload['objectId'] = inventories.objectId;
        }

        if (discoveryParameter == "unsigned_certificate") {
            // invoke Unsigned Certificates Discovery
            var peC = new PuppetEnterpriseUnsignedCerts(this.parameters, this.headers);
            var unsignedCertificates = peC.discoverUnsignedCert();
            discoveryPayload['unsignedCertificates'] = unsignedCertificates.unsigned_certs;
            discoveryPayload['objectId'] = unsignedCertificates.objectId;
        }

        if (discoveryParameter == "moduleClasses") {

        }
    }
```
puppet-enterprise-1.0-BootstrapNode

For the cloud API, BootstrapNode, use the following script:

```javascript
bootstrapNode();

function bootstrapNode()
{
    // install the puppet agent at host in cloud
    var invObj = new PuppetEnterpriseInventory(this.parameters, this.headers);
    var bootStrapNodePayload = invObj.installAgentAtNode();
    // return bootstrapNodePayload;
}
```

puppet-enterprise-1.0-DeregisterNode

For the cloud API, DeregisterNode, use the following script:

```javascript
var peI = new PuppetEnterpriseInventory(this.parameters, this.headers);
peI.unsignCertANode();
```

puppet-enterprise-1.0-ExecuteConfigPackages

For the cloud API, ExecuteConfigPackages, use the following script:

```javascript
// Compare Group Class parameters, if it is same, continue to create job ::
else update group parameter's value and create job.
var pc = new PuppetEnterpriseClassification(this.parameters, this.headers);
pc.compareGroupParameterValue();
var pj = new PuppetEnterpriseJob(this.parameters, this.headers);
pj.launchAJob();
```

puppet-enterprise-1.0-RegisterNode

For the cloud API, RegisterNode, use the following script:

```javascript
registerNode();
```
function registerNode() {
// accept certificate of host at Puppet Enterprise
    var invObj = new PuppetEnterpriseInventory(this.parameters,
                                          this.headers);
    var registerNodePayload = invObj.acceptCertRequest();

    //return registerNodePayload;
}
initialize: function(parameters, headers) {
  // define Macros
  this.puppetTargetRunCount = 50;
  this.puppetWaitPeriod = 20000;

  this.logger = new PuppetEnterpriseLogger();
  this.parameters = parameters;
  var ENDPOINT = "Endpoint";
  var USERNAME = "Identity";
  var PASSWORD = "Credentials";
  var CONFIG_MGMT_PROVIDER_INFO = "ConfigMgmtProviderInfo";
  this.DEFAULT_ERROR_STATUS_CODE = "500";
  this.PORT_CLOSED = "PORT-4433 CLOSED";
  this.URL_NOT_FOUND = "404";

  var configMgmtProvider =
    this.parameters.get(CONFIG_MGMT_PROVIDER_INFO);
  var configData = new JSON().decode(configMgmtProvider);
  this.enterpriseUrl = this.parameters.get(ENDPOINT);
  this.enterpriseUsername = configData.UserName;
  this.enterprisePassword = configData.Password;
  this.machine_username = configData.Machine_UserName;
  this.machine_password = configData.Machine_Password;
  this.objectId = configData.ObjectId;
},

_getNodeAddress: function(){
  var nodeAddressJson = this.parameters.get("NodeAddress");
  var nodeAddressJsonPayload = new JSON().decode(nodeAddressJson);
  var publicIp = nodeAddressJsonPayload.public_ip;
  var privateIp = nodeAddressJsonPayload.private_ip;
  var reachableIPAddress;
  if(!JSUtil.nil(publicIp))
    reachableIPAddress = publicIp;
  if(!JSUtil.nil(privateIp))
    reachableIPAddress = privateIp;
  return reachableIPAddress;
},

_getNodeName: function(){
  var additionalParams = this.parameters.get("AdditionalParameters");
  var additionalParamsJson = new JSON().decode(additionalParams);
  this.logger.info("***** Node Name from Additional Parameters :" +
    additionalParamsJson.node_name);
  return additionalParamsJson.node_name;
},
/*
 * To query Puppet Enterprise, token must be generated.
 */
_getToken: function() {
    var payload = {};
    payload["login"] = this.enterpriseUsername;
    payload["password"] = this.enterprisePassword;

    try {
        var authUrl = this.enterpriseUrl + ":4433/rbac-api/v1/auth/token";
        var request = new this.HTTPRequest(authUrl);
        request.addHeader("Content-Type", "application/json");
        var payloadJsonObj = new JSON().encodeObject(payload);
        var response = request.post(payloadJsonObj);
        var status = response.getStatusCode();
        var responseBody = response.getBody(); var jsonObj = new JSON().decode(responseBody);
        var token = jsonObj.token;

        return token;
    } catch (exception) {
        // ducking exception to main method
        throw new PuppetEnterpriseAPIException(exception.toString(),status);
    }
},

/**
 * Get Additional Parameters from route headers and parse it and return in a dictionary form
 */
_getAddtionalParameters: function(){
    var additionalParams =  this.parameters.get("AdditionalParameters");
    this.logger.info("***** Additional Parameters from CAPI header:
+additionalParams");
    var additionalParamsJson = new JSON().decode(additionalParams);
    var mgmtAttrsJson = additionalParamsJson.management_attributes;
    var additionalParameters = ();

    for(var outListIndex=0; outListIndex<mgmtAttrsJson.length; outListIndex++){
        var attrData = mgmtAttrsJson[outListIndex];
        var key = "";
        var value = "";
        for(var attr in attrData){
            if(attrData[attr].name == "management_attribute"){
                key = attrData[attr].value;
            } else if(attrData[attr].name == "management_attribute_value") {
                value = attrData[attr].value;
            }
        }
        additionalParameters[key] = value;
    }
    this.logger.info("***** Management Attributes from additional parameters:
+JSON.stringify(additionalParameters));
    return additionalParameters;
},

/**

* Get ConfigOverrides from route headers and parse it and return in a
dictionary form

```javascript
_getConfigOverrides: function() {
    var configAttrs = this.parameters.get("ConfigurationOverrides");
    var configAttrsJson = new JSON().decode(configAttrs);
    var configParameters = {};
    for(var outListIndex=0; outListIndex<configAttrsJson.length;
        outListIndex++){
        var attrData = configAttrsJson[outListIndex];
        var key="";
        var value="";
        for(var attr in attrData){
            if(attrData[attr].name == "attribute"){
                key = attrData[attr].value;
            }
            if(attrData[attr].name == "value"){
                value = attrData[attr].value;
            }
        }
        configParameters[key] = value;
    }
    this.logger.info("***** ConfigOverrides... :
    +JSON.stringify(configParameters));
    return configParameters;
}
```

/**
 * HTTP Responses are parsed.
 */
_getResponse: function(response) {
    var status = response.getStatusCode();
    var responseBody = response.getBody();
    if ((status < 200) || (status >= 300)) {
        throw new PuppetEnterpriseAPIException(responseBody, status);
    } else {
        var jsonObj = new JSON().decode(responseBody);
        return jsonObj;
    }
},

/**
 * HTTP GET Request are made to PuppetEnterprise and return responses.
 * This method takes api path as input parameter from calling method.
 */
_httpGET: function(apiPath) { try {
    //get Login token
    var token = this._getToken();
    var getUrl = this.enterpriseUrl + apiPath;
    var request = new this.HTTPRequest(getUrl);
    request.addHeader("Content-Type", "application/json");
    request.addHeader("X-Authentication", token);
    var response = request.get();
    var jsonObj = this._getResponse(response);
    return jsonObj;
} catch (exception) {
    //duckking exception to main method
    //Handling server error 500.
    ms.log('Error occurred while making get request '+ response);
}
throw new PuppetEnterpriseAPIException(exception.toString(), this.DEFAULT_ERROR_STATUS_CODE);
}

/**
 * HTTP POST Request are made to PuppetEnterprise and returns responses.
 * This method takes api path and payload as input parameter from calling method.
 */
_httpPOST: function(apiPath, payload) {
  try {
    var token = this._getToken();
    var postUrl = this.enterpriseUrl + apiPath;
    var request = new this.HTTPRequest(postUrl);
    request.addHeader("Content-Type", "application/json");
    request.addHeader("X-Authentication", token);
    var response = request.post(payload);
    var jsonObj = this._getResponse(response);
    return jsonObj;
  } catch (exception) {
    // duckking exception to main method
    throw new PuppetEnterpriseAPIException(exception.toString(), this.DEFAULT_ERROR_STATUS_CODE);
  }
},

/**
 * HTTP PUT Request are made to PuppetEnterprise and returns responses.
 * This method takes api path and payload as input parameter from calling method.
 */
_httpPUT: function(apiPath, payload) {
  try {
    var token = this._getToken();
    var putUrl = this.enterpriseUrl + apiPath;
    var request = new this.HTTPRequest(putUrl);
    request.addHeader("Content-Type", "application/json");
    request.addHeader("X-Authentication", token);
    var response = request.put(payload);
    var jsonObj = this._getResponse(response);
    return jsonObj;
  } catch (exception) {
    // duckking exception to main method
    throw new PuppetEnterpriseAPIException(exception.toString(), this.DEFAULT_ERROR_STATUS_CODE);
  }
},

  type: 'PuppetEnterpriseAPIBase'
};
* A runtime custom exception that is thrown in the course of an Ansible Tower API invocation.

```javascript
var PuppetEnterpriseAPIException = Class.create();
PuppetEnterpriseAPIException.prototype = Object.extend(new RESTAPIException(), {
  type: 'PuppetEnterpriseAPIException'
});
```

---

**PuppetEnterpriseClassification**

```javascript
var PuppetEnterpriseClassification = Class.create();
PuppetEnterpriseClassification.prototype = Object.extendObject(PuppetEnterpriseAPIBase, {
  initialize: function (parameters, headers) {
    this.apiPath = "/api/classifier/groups";
    PuppetEnterpriseAPIBase.prototype.initialize.call(this, parameters, headers);
  },

  discoverGroups: function () {
    this.logger.info("*** Puppet Enterprise Classification ***");
    var groupPayload = [];
    try {
      //Get group Lists
      var groupsResp = this._httpGET(this.apiPath);
      for (var index = 0; index < groupsResp.groups.length; index++) {
        //build group-table data for SNOW CMDB
        var groupDetails = {};
        var paramsList = [];
        groupDetails['name'] = groupsResp.groups[index].name;
        groupDetails['class_names'] = JSON.stringify(Object.keys(groupsResp.groups[index].classes));
        groupDetails['class_params'] = paramsList;
        groupPayload.push(groupDetails);
      }
    } catch (e) {
      console.log(e);
    }
    var groupData = {};
    groupData.groups = groupPayload;
    groupData.objectId = this.objectId;
  }
});
```
return groupData;
} catch (exception) {
    // ducking eindexception to main method
    throw exception;
},

/**
 * This method add a class to the specific classification
 * Inputs are from Stack variables and blueprint, which is invoked from
 * ExecuteConfig operation.
 */
compareGroupParameterValue: function () {
    this.logger.info("*** Compare Group parameter and Update at Puppet
Enterprise ***");
    try {
        var additionalParams = this._getAddtionalParameters();
        var configParams = this._getConfigOverrides();
        var configScriptJson = this.parameters.get("ConfigScript");
        var configScript = new JSON().decode(configScriptJson);
        var groupName = configScript['name'];
        var nodeName = this._getNodeName();
        this.logger.info("*** ConfigOverrides at
compareGroupParameterValue method: "+ JSON.stringify(configParams) + " with
Classification name : "+ groupName);
        // Get the details of the Group from the puppet enterprise
        var groupResp = this._httpGET(this.apiPath);
        var groupsList = groupResp.groups;
        var groupIndex = groupsList.map(function(o) { return
            o.name; }).indexOf(groupName);
        if(groupIndex < 0)
            throw new PuppetEnterpriseAPIException(" Classification Not
Found at Puppet Enterprise.. Please Re-try this operation after Discovery
operation");
        this.logger.info("*** Group Id "+groupsList[groupIndex].id + " from
Group name : "+ groupName);
        var groupClassUrl = this.apiPath + "/" +
            groupsList[groupIndex].id;
        var groupClassResp = this._httpGET(groupClassUrl);
        // a deepcopy of groupClass response
        var payload = JSON.parse(JSON.stringify(groupClassResp));
        // update class parameter value in classification
        for(var cName in groupClassResp.group.classes){
            for(var params in configParams){
                var className = params.split('~')[0];
                var paramName = params.split('~')[1];
                this.logger.info("*** Class Name: "+ className + "
Parameter Name: "+ paramName);
                if({cName == className) && (paramName in
                    groupClassResp.group.classes[cName]){)
                    // update at PE
                    payload.group.classes[cName][paramName] =
                        parseInt(configParams[params]);
var putPayload = new JSON().encode(payload);
var updatedGroupClassResp = this._httpPUT(groupClassUrl, putPayload);
this.logger.info("***** Update of Classification Response: "+JSON.stringify(updatedGroupClassResp));
}
}
}

var pinPayloadDict = {};
pinPayloadDict.nodes = [];
pinPayloadDict.nodes.push(this.nodeName.toLowerCase());

var pinPayload = new JSON().encode(pinPayloadDict);

var pinApiPath = ":4433/classifier-api/v1/groups/" + groupsList[groupIndex].id + "/pin";
var pinTogroupResp = this._httpPOST(pinApiPath, pinPayload);
this.logger.info("*** Added Node : "+ this.nodeName + " to Classification : "+ groupName);
}
}

PuppetEnterpriseCredentialResolve

var PuppetEnterpriseCredentialResolver = Class.create();

PuppetEnterpriseCredentialResolver.prototype = Object.extendObject(CAPIResolverBase, { 
  CredentialsProviderFactory: Packages.com.snc.commons.credentials.CredentialsProviderFactory,
  initialize: function(parameters) {
    CAPIResolverBase.prototype.initialize.call(this, parameters);
  },

  getServerCredentials: function() {
    var serverCredentials = {};
    var serverCredentialId = this.parameters.get('serverCredentialId');
    var object_id = this.parameters.get('object_id');
    if(serverCredentialId) {
      serverCredentials = this.getCredentials(serverCredentialId,object_id);
    } 
    return new JSON().encode(serverCredentials);
  },

  getCredentials: function(serverCredentialId,object_id) {
  }});
var credential = this.CredentialsProviderFactory.getCredentialsProvider().getCredentialByID(serverCredentialId);

var keyDetails = {
    UserName: credential.getAttributeByName("user_name"),
    Password: credential.getAttributeByName("password"),
    Machine_UserName: credential.getAttributeByName("machine_username"),
    Machine_Password: credential.getAttributeByName("machine_password"),
    ObjectId: object_id,
};

return keyDetails;

var credentialResolver = new PuppetEnterpriseCredentialResolver(this.parameters);
credentialResolver.getServerCredentials();

PuppetEnterpriseEnvironments

var PuppetEnterpriseEnvironments = Class.create();
PuppetEnterpriseEnvironments.prototype = Object.extendsObject(PuppetEnterpriseAPIBase, {
    initialize: function (parameters, headers) {
        this.apiPath = "/api/classifier/environments";
        PuppetEnterpriseAPIBase.prototype.initialize.call(this, parameters, headers);
    },

    /**
     * This method Discovers Environment at Puppet Enterprise upon Discover operation.
     * updates CMDB tables data
     */
    discoverEnvironments: function () {
        this.logger.info("*** PUPPET ENVIRONMENTS ***");
        var environmentPayload = [];

        try {
            var envnRep = this._httpGET(this.apiPath);
            for (var index = 0; index < envnRep.length; index++) {
                var environmentDetails = {};
                environmentDetails['name'] = envnRep[index];

                environmentPayload.push(environmentDetails);
            }

            var environmentData = {};
            environmentData.environments = environmentPayload;
            environmentData.objectId = this.objectId;
            return environmentData;
        } catch (e) {
            this.logger.error(e);
        }
    }
});
try {
    // Get Environments Lists
    var environList = this.discoverEnvironments();
    var modulesPayload = [];

    // Loop over each environment, to get modules
    for(var index=0; index < environList.environments.length; index++) {
        var moduleClassUrl = this.apiPath + environList.environments[index].name + '/class-names';
        var moduleClassResp = this._httpGET(moduleClassUrl);

        // Loop each modules to get parameters
        for(var moduleindex=0; moduleindex < moduleClassResp.length; moduleindex++) {
            var moduleUrl = this.apiPath + environList.environments[index].name + '/classes/' + moduleClassResp[moduleindex];
            var moduleClassParamResp = this._httpGET(moduleUrl);
            var modulesData = {};
            modulesData.name = moduleClassResp[moduleindex];
            modulesData.parameters = moduleClassParamResp.parameters;
            modulesPayload.push(modulesData);
        }
    }

    var moduleData = {};
    moduleData.moduleClasses = modulesPayload;
    moduleData.objectId = this.objectId;
    return moduleData;
} catch (exception) {
    // duckking exception to main method
    throw exception;
}
PuppetEnterpriseInventory

```javascript
var PuppetEnterpriseInventory = Class.create();
PuppetEnterpriseInventory.prototype = Object.extendsObject(PuppetEnterpriseAPIBase, {

  initialize: function (parameters, headers) {
    this.apiPath = "/api/classifier/nodes";
    this.secureShellhelper = new this.SecureShellHelper();
PuppetEnterpriseAPIBase.prototype.initialize.call(this, parameters, headers);
  },

  /*
   * This method is to Accept Cert Request from Puppet Agent at Puppet Enterprise upon Register Node operation.
   */
  acceptCertRequest: function () {
    this.apiPath = "/api/certificates/nodes";
    var additionalParameters = this.parameters.get('AdditionalParameters');
    var additionalParametersPayload = new JSON().decode(additionalParameters);
    var nodeName = additionalParametersPayload['node_name'].toLowerCase();
    var count = 0;
    var currentState = "Notrequested";
    var proxyInfo = this.parameters.get('mid.capi.proxyInfo');
    do{
      this.logger.info("***** Starting invocation of Certificate Acceptance at PE with Additional Parameters at acceptCertRequest method: " + additionalParameters + " with NodeName: " + nodeName + " with Count: " + count);

      var certListResp = this._httpGET(this.apiPath);
      var certList = certListResp.nodes;
      var index = certList.map(function (o) { return o.name; }).indexOf(nodeName);
      this.logger.info("**** Certificate for Node at Puppet Enterprise: " + JSON.stringify(certList[index]));

      if((certList[index].name == nodeName) && (certList[index].state == "requested")){
        currentState = certList[index].state;
        this.logger.info("*** Node Certificate Status: " + currentState);
        //accept the request
        var payload = {};
        payload.nodes = [];
        payload.nodes.push(nodeName);
        payload.action = "accept";

        var postPayload = new JSON().encode(payload);
        this.acceptCertResp = this._httpPOST(this.apiPath, postPayload);
      }

      //update current state
      currentState = this.acceptCertResp.nodes[0].state;

      count ++;
      this.logger.info("***** Starting invocation of Certificate Acceptance at PE with Additional Parameters at acceptCertRequest method: " + additionalParameters + " with NodeName: " + nodeName + " with Count: " + count);
    }
```
this.logger.info("*** Node Accept Certificate Response:
"+JSON.stringify(this.acceptCertResp));

} else {
    this.logger.info("*** Waiting for the cert to be in the
requested state ***, since current state is "+currentState + ", re-try
count: " + count);
    //sleep for 5 secs
    this.Thread.sleep(this.puppetWaitPeriod);
    } count ++;
}while (( currentState != "signed") && (count <this.puppetTargetRunCount ));

if((currentState == "failed")){
    this.logger.info("*** Certificate Request Not Accepted at PE");
    throw new PuppetEnterpriseAPIException(this.acceptCertResp.nodes[0].error.details,
    this.acceptCertResp.nodes[0].error.status);
} if((count >= this.puppetTargetRunCount) && (currentState ==
"Notrequested")){
    this.logger.info("*** Certificate Request Not Accepted at PE
Since Maximum re-try count reached ***");
    throw new PuppetEnterpriseAPIException("**** Certificate Request
Not Accepted at PE Since Maximum re-try count reached" , "404");
} this.logger.info('Status after Accept Cert Request Operation:
"+currentState);

}/**
 * This method Discovers List of Inventories at Puppet Enterprise upon
Discover operation.
 * updates CMDB tables data
 */
discoverInventories: function () {
    this.logger.info("*** PUPPET INVENTORY ***");
    var inventoryPayload = [];

    var FactRep = "";

    try {
        //Get Inventory Lists
        var payload = {};
        var invenResp = this._httpPOST( this.apiPath, payload);

        for (var index = 0; index < invenResp.length; index++) {
            var inventoryDetails = {};
            inventoryDetails['name'] = invenResp[index].certname;
            inventoryDetails['certname'] = invenResp[index].certname;
            inventoryDetails['catalogenvironment'] =
            inventoryDetails['reportenvironment'] =
            inventoryDetails['factsenvironment'] =
            inventoryDetails['latest_report_status'] =
            inventoryDetails['cached_catalog_status'] =
            invenResp[index].cachedCatalogStatus;
inventoryDetails['latest_report_noop_pending'] = invenResp[index].latestReportNoopPending;
inventoryDetails['latest_report_noop'] = invenResp[index].latestReportNoop;
inventoryDetails['latest_report_corrective_change'] = invenResp[index].latestReportCorrectiveChange;
inventoryDetails['latest_report_hash'] = invenResp[index].latestReportHash;

inventoryPayload.push(inventoryDetails);

var nodeId = invenResp[index].id;
var factUrl = "/api/classifier/nodes/" + nodeId + "/facts";
var factObj = new PuppetEnterpriseFacts(this.parameters,
this.headers);
FactRep = factObj.discoverFacts(nodeId, factUrl);

var inventoryData = {};
inventoryData.inventories = inventoryPayload;
inventoryData.facts = FactRep;
inventoryData.objectId = this.objectId;
return inventoryData;

} catch (exception) {
    // ducking exception to main method
    throw exception;
}


/*
 * This method installs Puppet Agent at node using SecureShell class
 * Shell Script is the input from blue print
 */
installAgentAtNode: function(){
    this.logger.info("*** INSTALL PUPPET AGENT AT NODE ***");
    var installCounter=0;

    try{
        var additionalParameters = this.parameters.get('AdditionalParameters');
        var additionalParametersPayload = new JSON().decode(additionalParameters);
        var configScript = this.parameters.get('ConfigScript').toString();
        var nodeAddress = this._getNodeAddress();
        var nodeCredentialsJson = this.parameters.get('NodeCredentials');
        var nodeCredential = new JSON().decode(nodeCredentialsJson);
        var userName=nodeCredential['UserName'];
        var password=nodeCredential['Password'];
        var privateKey=nodeCredential['PrivateKey'];
        var privateKeyValid =privateKey.replaceAll('\n',\n);  
        var shellResObj = "";
        var proxyInfo = this.parameters.get('mid.capi.proxyInfo');
        do{
            var shellRes = this.secureShellhelper.executeCommand(nodeAddress, '22', userName,
password, privateKey, privateKeyValid, configScript, "sh", false, true,
true,proxyInfo);

            // this needs to be modified.
            this.logger.info("*** Execution response for script
execution object " +shellRes);
//parse shell response for exitcode and status
shellRes = shellRes.toString().split('ExecutionResponse')[1];
    shellRes = shellRes.replaceAll('=','
      ').replaceAll('stdOut', '"
      stdOut"
      ') .replaceAll('stdErr','"stdErr"
      ').replaceAll('exitCode','"exitCode"
      ').replaceAll('executedOn','"executedOn"
      ').replaceAll('response','"response"
      ') .replaceAll("\n" , " ");

//if length of shell response is less than 500, we assume
its an abnormal message from shell
if (shellRes.length() < 500){
    shellResObj = JSON.parse(shellRes);
    this.logger.info(" Install ShellScript Exit Code : "+shellResObj.exitCode
    );

    //exitCode = -1 means system is set to reboot.
    if((shellResObj.exitCode == -1) && (this.count <3)){
        //sleep for few seconds, so that agent will request
        master
        this.logger.info("***** Sleep for a while!!! Before
        we initiate Re-installation of Puppet Agent at Node *****");
        //sleep for 5 sec
        this.Thread.sleep(this.puppetWaitPeriod);
        this.logger.info("***** Woke-up from sleep!!! Ready
        to Re-installation of Puppet Agent at Node: " + this.nodeName + " with
        install Count: " + installCounter);
    }
    installCounter++;
}
} while((shellResObj.exitCode == -1) && (installCounter <3));

if((shellResObj.exitCode == -1) && (installCounter >= 3)){
    throw new PuppetEnterpriseAPIException("VM Provisioned for
    PuppetEnterprise doesnot Configured with mandatory data" , "400");
}
}

try (exception) {
    //duckking exception to main method
    throw exception;
}
,

/*
* This method Unsignes the ssh/ssl Certificate of node at Puppet
* Enterprise, which is invoked at De-registernode operation
*/

unsignCertANode: function(){
    // code to unsign certificate
    this.logger.info("**** Unsigning Certificate for Node ****");
    try{
        var nodeName = this._getNodeName().toLowerCase();
        var configScript =
        this.parameters.get('ConfigScript').toString();
        var nodeAddress = this.enterpriseUrl.split("/")[1];
        var userName= this.machine_user;
        var password= this.machine_password;
        var shellRes = ";
        var proxyInfo = this.parameters.get('mid.capi.proxyInfo');
```javascript
this.logger.info("*** Unpin the Node from all Classification it has connected ***");
var unpinApiPath = ":4433/classifier-api/v1/commands/unpin-from-all";
var unpinPayloadDict = {};
unpinPayloadDict.nodes = [];
unpinPayloadDict.nodes.push(nodeName.toLowerCase());
var unpinPayload = new JSON().encode(unpinPayloadDict);
var unpinResp = this._httpPOST(unpinApiPath,unpinPayload);
this.logger.info("***unpin Resp :"+JSON.stringify(unpinResp));
this.logger.info("*** Remove/Purge Node from Puppet Enterprise ***");

if(userName == "root"){
    shellRes =
    this.secureShellhelper.executeCommand(nodeAddress, '22', userName, password, ", configScript, "sh", false, false, false,proxyInfo);
} else {
    shellRes =
    this.secureShellhelper.executeCommand(nodeAddress, '22', userName, password, ", configScript, "sh", false, true, true,proxyInfo);
}

shellRes = shellRes.toString().split('ExecutionResponse')[1];
this.logger.info(" UncertOfNode Response: "+ shellRes);

})catch (exception) {
    //duckking exception to main method
    throw exception;
},

}),

```

**PuppetEnterpriseJob**

```javascript
var PuppetEnterpriseJob = Class.create();
PuppetEnterpriseJob.prototype =
Object.extendsObject(PuppetEnterpriseAPIBase, {

initialize: function(parameters, headers) {
    this.apiPath = "/api/jobs/";
    this.envObj = new PuppetEnterpriseEnvironments(parameters, headers);
    PuppetEnterpriseAPIBase.prototype.initialize.call(this, parameters, headers);
},

/**
 * This method fires a job template at Puppet Enterprise.
 * Inputs are from Stack variables and blueprint, which is invoked from ExecuteConfig operation.
 */
launchAJob: function(){
    this.logger.info("*** Launching a Job at Puppet Enterprise ***");
    try{
        //initialize NodeList with list of nodes if needed, else a single node on which Job has to run.
        var jobFlag = "";
```
var pcpStatusCount=0;

//get certname of agent as registered at puppet enterprise
this.nodeName = this._getNodeName().toLowerCase();

do{
    //sleep for some seconds, so that puppet agent gets connected to PCP Broker, to Run a job.
    //sleep for 5 sec
    this.pcpStatus = this.getStatusOfPCPBrokerForANode(this.nodeName);
    this.logger.info("***** Run puppet status for node " + this.nodeName + " in inventory is still: " + this.pcpStatus +" Lets keep waiting....with Re-try Count: " + pcpStatusCount);
    this.Thread.sleep(this.puppetWaitPeriod);
    pcpStatusCount++;
}while((this.pcpStatus != true) && (pcpStatusCount < this.puppetTargetRunCount));

this.logger.info("***** Wait completed for node " + this.nodeName + ' Now ready to run job '+ this.pcpStatus);
if(this.pcpStatus == true){
    //payload for HTTP POST
    var payload = {};
    payload.certname = this.nodeName;
    payload.flags = [];

    var postPayload = new JSON().encode(payload);
    var launchAPIPath = "/api/agents/run";
    var jobLaunchResp = this._httpPOST(launchAPIPath, postPayload);
    this.logger.info("***** Job Launch Response : "+ JSON.stringify(jobLaunchResp ));
    //check status of job launched.
    var jobAPIPath = this.apiPath+jobLaunchResp.transactionId ;
    this.checkJobStatus(jobAPIPath);
}
else{
    throw new PuppetEnterpriseAPIException('PCP Broker of Agent is not connected to your Puppet Enterprise, Please check your Puppet Enterprise/Node configuration ',this.DEFAULT_ERROR_STATUS_CODE);
}
}

try{
    var jobStatusCount = 0;
    var jobResp = "";
    do{

jobResp = this._httpGET(apiPath);

if(jobResp.job.state.toString() == "finished") {
  this.logger.info(" JobId: "+ jobResp.job.id +" Launch status : "+jobResp.job.state);
  return true;
}

else if(jobResp.job.state.toString() == "failed") {
  //check Job status logs , for error details
  var jobStatusLogAPI = apiPath + "/nodes";
  var jobStatusLogResp = this._httpGET(jobStatusLogAPI);

  for(var index=0; index < jobStatusLogResp.nodes.length; index++){
    if((jobStatusLogResp.nodes[index].name == this.nodeName.toLowerCase()) && (jobStatusLogResp.nodes[index].state == "errored")){
      throw new PuppetEnterpriseAPIException('Job Execution Failed at Puppet Enterprise with Message: ' + jobStatusLogResp.nodes[index].message,this.DEFAULT_ERROR_STATUS_CODE);
    }
  }
}

else if(jobResp.job.state.toString() == "running"){
  //sleep for 5 secs
  this.logger.info("***** Sleep for a while!!! Since Job is Running *****");
  this.Thread.sleep(this.puppetWaitPeriod);
}

else{
  //throw exception for unknown state of a job
  throw new PuppetEnterpriseAPIException('Job Execution State is unpredictable at Puppet Enterprise with state: ' + jobResp.job.state.toString(),this.DEFAULT_ERROR_STATUS_CODE);
}

jobStatusCount++;
}

while((jobResp.job.state.toString() != "finished") && (jobStatusCount < this.puppetTargetRunCount));

//throw exception for unknown state of a job
if((jobResp.job.state.toString() != "finished") && (jobStatusCount >= this.puppetTargetRunCount))
  throw new PuppetEnterpriseAPIException('Job Execution State is unpredictable at Puppet Enterprise with state: ' + jobResp.job.state.toString(),this.DEFAULT_ERROR_STATUS_CODE);

}

} catch(exception){
  //duckking exception to main method
  throw exception;
}

/**
 * This method gets status of PCP (Puppet Communications Protocol) broker for a node
 * This method is invoked from launchAJob method
 * Inputs are from launchAJob, which is Node Cert Name.
getStatusOfPCPBrokerForANode: function(nodeCertName) {
    try {
        var pcpApiPath = ":8143/orchestrator/v1/inventory";
        var payload = {};
        payload.nodes = [nodeCertName];
        var pcpPostPayload = new JSON().encode(payload);
        var pcpResp = this._httpPOST(pcpApiPath, pcpPostPayload);

        this.logger.info("***** PCP Post Response for Node : " + nodeCertName + ": resp: " + JSON.stringify(pcpResp));
        for (var index = 0; index < pcpResp.items.length; index++) {
            if (pcpResp.items[index].name == nodeCertName) {
                return pcpResp.items[index].connected;
            }
        }
    }
    catch (exception) {
        // ducking exception to main method
        throw exception;
    }
},

PuppetEnterpriseLogger

var PuppetEnterpriseLogger = Class.create();
PuppetEnterpriseLogger.prototype = { StringUtil: Packages.com.glide.util.StringUtil,

    initialize: function() {
        this.shouldLogDebug = ('true' ==
            this._getMIDProperty('mid.cloudapi.debug', 'false'));
    },

debug: function(msg) {
    ms.log('*** DEBUG: (' + this.type + '): ' + msg);
},

info: function(msg) {
    ms.log('*** INFO: (' + this.type + '): ' + msg);
},

warn: function(msg) {
    ms.log('*** WARNING: (' + this.type + '): ' + msg);
},

error: function(msg) {
    ms.log('*** ERROR: (' + this.type + '): ' + msg);
},

    /**
   * Generate a string from current date/time in the format
   * YYYYMMDDHHMMSSMMM
   */
    _getDateStr: function() {
        var date = new Date();
        var dateString = date.getFullYear() +
            ('0' + (date.getMonth() + 1)).slice(-2) +
            ('0' + date.getDate()).slice(-2) +
            ('0' + date.getHours()).slice(-2) +
            ('0' + date.getMinutes()).slice(-2) +
            ('0' + date.getSeconds()).slice(-2) +
            (millis / 1000).toString().slice(0, 3);
        return dateString;
    },
Create a CI class for an environment

Create a configuration item (CI) class for an environment and add identification rules to the CI class. Creating a class and adding identification rules makes managing the CIs more efficient. An environment can be a production, development, or a test environment.

Role required: sn_cmp.cmp_root_admin

1. Navigate to Configuration > CI Class Manager.
2. To display the CI classes list, click Hierarchy.
3. Search for and open the Config Entity (cmdb_ci_config_entity) CI class.
4. To create a new child class, and move to the Provide Basic Info stage, click Add Child Class.
5. In the Provide Basic Info stage, enter a name (Puppet Enterprise Environment) for the class in the Display Name field.
6. Click Next until you reach the Set Identification Rule stage.
7. Create an identification rule.
   a) Click Add.
      The Create Identification Rule dialog box appears.
   b) Click Dependent and click Save.
   c) To create an identifier entry, click Add in the Identifier Entries section.
   d) Move Object ID to the Selected list.
   e) Click Save.

A class is created for the environment and identification rules have been created.

Configure Metadata Rules for Puppet provider

Configure metadata rules to create a relationship between the Puppet enterprise CI class and the Puppet enterprise environment CI class.
Role required: sn_cmp.cmp_root_admin

1. Navigate to Metadata Editor.
2. Click Hosting Rules and then click Add New Rule.
   Add Top Level Hosting Rule dialog box appears.

3. Select Puppet Enterprise Environment from the Configuration Item Type list box and click Create
   Puppet Enterprise Environment is listed in the Metadata Editor window.
4. Search for Puppet Enterprise Environment and click the plus icon (Add hosting rule) next to it.
   The Add Hosting Rule to Puppet Enterprise Environment dialog box appears.
5. Select Puppet Enterprise from the Configuration Item Type list box.
6. De-select the Reverse Relationship Direction check box. By default, this check box is enabled.
7. Click Create.

Create a resource block for Puppet provider

Create a resource block for Puppet provider
Create a resource block for the Puppet provider based on a CI type from the CMDB.

Role required: sn_cmp.cmp_root_admin

1. In the Cloud Admin Portal, navigate to Design > Resource Blocks.
2. Click New.
   The Create Resource Block dialog box appears.
3. Fill in the form fields (as shown in the table). For a detailed description of each field, see Create a custom resource block.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (Puppet Enterprise) for the resource block.</td>
</tr>
<tr>
<td>Layer</td>
<td>Select the type of resource block (Virtual Infrastructure).</td>
</tr>
<tr>
<td>CI Type</td>
<td>Select the CI type (Puppet Enterprise) that the resource block represents.</td>
</tr>
<tr>
<td>Guest Interface</td>
<td>Select a default guest interface (Puppet Enterprise Interface) to use when you add a resource block to a blueprint.</td>
</tr>
</tbody>
</table>
4. Click **Submit**.
   The resource block information appears.

5. Click the **Operations** sub tab and then click the plus icon to add an operation.
   The Add Operation dialog box appears.

6. Fill in the form fields (as shown in the table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create Interface</td>
<td>Select the check box to create a custom guest interface.</td>
</tr>
</tbody>
</table>

7. Click **Submit**.

8. Click the **Edit Operation** icon (next to the **Operation** list box) to add operation parameters.
   The Edit Operations dialog box appears.

9. Click the plus icon next to **OperationParameters**.

10. Enter the **Name** and **Type** parameters as mentioned in the table.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CfgMgmtEndpoint</td>
<td>String</td>
</tr>
<tr>
<td>ConfigMgmtProviderInfo</td>
<td>String</td>
</tr>
<tr>
<td>WorkloadConfigProvider</td>
<td>String</td>
</tr>
</tbody>
</table>

11. Click **Close**.
    You need to add input parameters for the Discover operation.

12. Click the plus icon next to **Generate Catalog**.

13. Enter parameters and their corresponding mappings. Please ensure that you enter the exact mapping values as shown in the table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>CfgMgmtEndpoint</td>
<td>${ci.cmdb_ci_workload_config_provider($parameter.WorkloadConfigProvider)}</td>
</tr>
<tr>
<td>ConfigMgmtProviderInfo</td>
<td>${parameter.ConfigMgmtProviderInfo}</td>
</tr>
<tr>
<td>WorkloadConfigProvider</td>
<td>${ci.cmdb_ci_workload_config_provider($parameter.WorkloadConfigProvider)}</td>
</tr>
</tbody>
</table>

14. Click **Save Changes** (next to **Generate Catalog**).

15. Click **Steps** and then click the plus icon to add operation steps to discover environments.
    The Add Operation Steps dialog box appears.

16. Fill out the form fields (as shown in the table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td>Select <strong>Invoke Cloud API</strong> from the list.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>API Provider</td>
<td>Select <strong>Puppet Enterprise</strong> from the list.</td>
</tr>
<tr>
<td>API Interface</td>
<td>Select <strong>Configuration Management Interface</strong> from the list.</td>
</tr>
<tr>
<td>API Method</td>
<td>Select <strong>Discover</strong> from the list.</td>
</tr>
<tr>
<td>CAPI Version</td>
<td>Enter the value <strong>1.0</strong> in the field.</td>
</tr>
</tbody>
</table>

17. Click **Submit**.
18. For the parameter, **AdditionalParameters**, enter
   ```json
   {
   "organization": "${parameter.Organization}"
   },
   "entity": "environment"
   ```
in the **Value** field.
19. Click the **Resource Script** sub tab and then click **New**.
20. Enter a name, type, and define a script and then click **Submit**. You need to define the script titled, **Discover_Puppet_Environments**. See **Resource Scripts for Puppet Provider** located at the end of this topic.
21. Click the **Operations** sub tab and click **Steps**.
22. Click **Response Processor** and then click the plus icon. The Add Response Processor dialog box appears.
23. Select **Discover_Puppet_Environment** from the **Script Name** list and click **Submit**.
24. Click the plus icon again to add operation steps to discover groups. Use the same values that you used in step 16.
25. For the parameter **AdditionalParameters**, enter
   ```json
   {
   "organization": "${parameter.Organization}"
   },
   "entity": "group"
   ```
in the **Value** field.
26. Click the **Resource Script** sub tab and then click **New**.
27. Enter a name, type, and define a script and then click **Submit**. You need to enter two resource scripts (**Discover_Puppet_Config_Installables** and **Discover_Puppet_Create_Application_Template**). See **Resource Scripts for Puppet Provider** located at the end of this topic.
28. Click the **Operations** sub tab and click **Steps**.
29. Click **Response Processor** and then click the plus icon. The Add Response Processor dialog box appears.
30. Select **Discover_Puppet_Config_Installables** from the **Script Name** list and click **Submit**.
31. Click the plus icon again to enter another response processor. Select **Discover_Puppet_Create_Application_Template** from the **Script Name** list and click **Submit**.
32. Reset the order of the response processors as shown in the table and click **Save Changes**.

<table>
<thead>
<tr>
<th>Script Name</th>
<th>Execution Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discover_Puppet_Config_Installables</td>
<td>0</td>
</tr>
<tr>
<td>Discover_Puppet_Create_Application_Template</td>
<td></td>
</tr>
</tbody>
</table>

**Create cloud script templates for Puppet provider**

Resource scripts for Puppet provider

Several sample resource scripts are provided, by default, by the Cloud Management application for the Puppet provider.
Discover_Puppet_Environments

function processResponse(response, cloudServiceAccountId, ldc) {
    var PuppetModelString = [];
    var discoveryResponse = global.JSON.parse(response);
    gs.info("^*^*^uppet Environment Discovery Response ^*^*^"+response);
    var respIndex = 0;
    var tableName = "";

    //environment
    var environmentResponse = discoveryResponse.environments;
    for(respIndex= 0, environments_size = environmentResponse.length;
        respIndex <environments_size ; respIndex++){
        //hosting table = Enterprise for Environment
        tableName = getPuppetTableName("Puppet Enterprise Environment");
        var hostingTable = {};
        hostingTable.name = getPuppetTableName("Puppet Enterprise");
        hostingTable.identifier = "object_id";
        hostingTable.identifierValue = discoveryResponse.objectId;

        var envObjectId = discoveryResponse.objectId + "_"+
        environmentResponse[respIndex].name;
        var environmentTable = new BuildTableData(tableName,envObjectId,
        hostingTable);
        //add attributes for the table
        environmentTable[tableName].attributes.name =
        environmentResponse[respIndex].name;
        PuppetModelString.push(environmentTable);
    }
    return global.JSON.stringify(PuppetModelString);
}

function BuildTableData(tableName, id, hostingTable){
    this[tableName] = {};
    this[tableName].validator = "PuppetEnterpriseResponseValidator" ;
    this[tableName].validator_overrides = {};
    this[tableName].identification = {};
    this[tableName].identification [tableName] = {};
    this[tableName].identification[tableName].criterion= {};
    this[tableName].identification[tableName].criterion.object_id = id.toString();

    //attach hosting table identification
    this[tableName].identification[hostingTable.name] =
    buildTableIdentification(hostingTable);
    this[tableName].attributes = {};
}

function buildTableIdentification(hostingTable){
    var identification = {};
    identification[hostingTable.name] = {};
    identification[hostingTable.name].criterion = {};
    identification[hostingTable.name].criterion[hostingTable.identifier] =
    hostingTable.identifierValue;
    return identification[hostingTable.name];
}

function getPuppetTableName(name){
    var dbObjectGr = new GlideRecord('sys_db_object');
if(!dbObjectGr.isValid())
    gs.error('##### Puppet### Discover_Puppet_Create_Application_Template### failed to get table = ' + name );
    if(dbObjectGr.get('label', name)){
        return dbObjectGr.getValue('name');
    }
    return '';}

Discover_Puppet_Config_Installable

function processResponse(response, cloudServiceAccountId, ldc) {
    var PuppetModelString = [];
    var discoveryResponse = global.JSON.parse(response);
    gs.info("##### Rachana #### Puppet### Classification###-->"+response);
    var respIndex = 0;
    var tableName = "";

    //groups
    var groupResponse = discoveryResponse.groups;
    for(respIndex= 0, group_size = groupResponse.length; respIndex < group_size ; respIndex++){
        var hostingTables = [];

        //hosting table = Enterprise for config_installable
        tableName = "sn_cmp_cfg_installable";
        var hostingTable = {};
        hostingTable.name = getPuppetTableName("Puppet Enterprise");
        hostingTable.identifier = "object_id";
        hostingTable.identifierValue = discoveryResponse.objectId;
        hostingTables.push(hostingTable);

        var groupObjectId = discoveryResponse.objectId + "_"+
            groupResponse[respIndex].name;
        var configInstallTable = new BuildTableData(tableName,groupObjectId,
            hostingTables);
        //add attributes for the table
        configInstallTable[tableName].attributes.name =
            groupResponse[respIndex].name;
        configInstallTable[tableName].attributes.run_list =
            groupResponse[respIndex].class_names;
        PuppetModelString.push(configInstallTable);

        //payload for installableParameters
        var parameters = groupResponse[respIndex].class_params;//it is a
            List of dict's with key-value pair
        if(parameters.length >0){
            for(var index=0; index < parameters.length; index++){
                for(var param in parameters[index]){ gs.info("##### param-->"+param);
                    //hosting table = sn_cmp_cfg_installable
                    if(param in parameters[index] { configInstallables
                        var installParamTable = {};
                        installParamTable.name = "sn_cmp_cfg_installable";
                        installParamTable.identifier = "object_id";
                        installParamTable.identifierValue = groupObjectId;
                        hostingTables.push(installParamTable);}}
gs.info("### hostingTables---
"+JSON.stringify(hostingTables));
var paramtableName = "sn_cmp_cfg_installable_param";
var paramObjectId = groupObjectId +"_"+param;
var installableParamTable = new BuildTableData(paramtableName, paramObjectId, hostingTables );

//add attributes for table
installableParamTable[paramtableName].attributes.name = param;
installableParamTable[paramtableName].attributes.value = parameters[index][param];
PuppetModelString.push(installableParamTable);
gs.info("#### installableParamTable-->"+JSON.stringify(installableParamTable));
}
}
}
}
}
gs.info("#### PuppetModelString-->"+JSON.stringify(PuppetModelString));
return global.JSON.stringify(PuppetModelString);
}
function BuildTableData(tableName, id, hostingTables){
    this[tableName] = {};
    this[tableName].validator = "PuppetEnterpriseResponseValidator" ;
    this[tableName].validator_overrides = {};
    this[tableName].identification = {};
    this[tableName].identification [tableName] = {};
    this[tableName].identification[tableName].criterion= {};
    this[tableName].identification[tableName].criterion.object_id = id.toString();

    //attach hosting table identification
    for(var index in hostingTables){
        //attach hosting table identification
        this[tableName].identification[hostingTables[index].name] = buildTableIdentification(hostingTables[index]);
    }
    this[tableName].attributes = {};
}
function buildTableIdentification(hostingTable){
    var identification = {};
    identification[hostingTable.name] = {};
    identification[hostingTable.name].criterion = {};
    identification[hostingTable.name].criterion[hostingTable.identifier] = hostingTable.identifierValue;
    return identification[hostingTable.name];
}

function getPuppetTableName(name){
    var dbObjectGr = new GlideRecord('sys_db_object');
    if(!dbObjectGr.isValid())
        gs.error("##### Puppet### Discover_Puppet_Create_Application_Template### failed to get table = ' + name");
    if(dbObjectGr.get('label', name)){
        return dbObjectGr.getValue('name');
    }
Discover_Puppet_Create_Application_Template

function processResponse(response, cloudServiceAccountId, ldc) {
    var puppetEnterpriseTableLabel = 'Puppet Enterprise';
    var PuppetModelString = [];
    var discoveryResponse = global.JSON.parse(response);
    gs.info('Discover_Puppet_Create_Application_Template --> '+response);
    var respIndex = 0;
    //groups
    var groupResponse = discoveryResponse.groups;
    var puppetEnterpriseTableName =
        getPuppetTableName(puppetEnterpriseTableLabel);
    var providerInstance =
        getProviderInstance(puppetEnterpriseTableName,
            discoveryResponse.objectId);
    gs.info('Discover_Puppet_Create_Application_Template --> puppetEnterpriseTableName ' + puppetEnterpriseTableName + '  groupResponseLength = ' +
        groupResponse.length);
    for(respIndex= 0, group_size = groupResponse.length; respIndex <
        group_size ; respIndex++){
        gs.info('Discover_Puppet_Create_Application_Template --> respIndex ' + respIndex);
        //hostingTable.name = puppetEnterpriseTableName;
        var groupObjectId = discoveryResponse.objectId + '_'+
            groupResponse[respIndex].name;
        var installable = getInstallable( groupObjectId);

        createOrUpdateTemplates( installable.sys_id, installable.name, 
            groupObjectId, providerInstance.name, providerInstance.sys_id);
    }
    gs.info('Discover_Puppet_Create_Application_Template -->
        return "[]";')
}

function getInstallable( objectId){
    gs.info('Discover_Puppet_Create_Application_Template --> getInstallable ' +
        objectId );
    var installable = {};
    var installableGr = new GlideRecord('sn_cmp_cfg_installable');
    if(installableGr.get('object_id', objectId)){
        installable.name = installableGr.getValue('name');
        installable.sys_id = installableGr.getUniqueValue();
        installable.run_list = installableGr.getValue('run_list');
    }
    gs.info('Discover_Puppet_Create_Application_Template -->
        return installable;')
}

function createOrUpdateTemplates( configSysId, name, objectId,
    instanceName, providerInstanceId){
```javascript
var applicationTemplateGr = new GlideRecord('sn_cmp_application_template');
var templateId = getUniqueTemplateName(instanceName, objectId);
applicationTemplateGr.get('template_id', templateId);
applicationTemplateGr.setValue('name', name);
applicationTemplateGr.setValue('config_runlist_provider', 'sn_cmp_cfg_installable');
applicationTemplateGr.setValue('config_runlist', configSysId);
applicationTemplateGr.setValue('provider_instance', providerInstanceId);
applicationTemplateGr.setValue('template_id', templateId);
applicationTemplateGr.update();
}

function getProviderInstance(tableName, objectId){
    var providerInfo = {};
    var providerGr = new GlideRecord(tableName);
    if(providerGr.get('object_id', objectId)){
        providerInfo.objectId = objectId;
        providerInfo.sys_id = providerGr.getUniqueValue();
        providerInfo.name = providerGr.getValue('name');
    }
    return providerInfo;
}

function getUniqueTemplateName(instanceName, objectId){
    var templateName = 'template' + instanceName + objectId;
    return templateName.replace(/ /g, '');
}

function getPuppetTableName(name){
    var dbObjectGr = new GlideRecord('sys_db_object');
    if(!dbObjectGr.isValid())
        gs.error('##### Puppet### Discover_Puppet_Create_Application_Template### failed to get table = ' + name);
    if(dbObjectGr.get('label', name)){
        return dbObjectGr.getValue('name');
    }
    return '';  
}
```

Create cloud script templates for Puppet provider

Create cloud script templates to install a puppet agent on a virtual machine once the virtual machine is provisioned.

Role required: sn_cmp.cmp_root_admin

1. In the Cloud Admin Portal, navigate to Design > Cloud Scripts > Cloud Script Templates.
2. Click New. Create two script templates (as shown in the table).
<table>
<thead>
<tr>
<th>Name of the script</th>
<th>Script</th>
</tr>
</thead>
</table>
| **InstallPuppetAgent** | ```bash
#!/bin/sh
MASTER_ENDPOINT=#CfgMgmtEndpoint#
sed -i '/search/d' /etc/resolv.conf
curl -k "$MASTER_ENDPOINT:8140/packages/current/install.bash"| sudo bash
CMD="export PATH=$PATH:/opt/puppetlabs/puppet/bin:/opt/puppetlabs/puppet/lib:/opt/puppetlabs/puppet/bin/gem"
CMD="service puppet start"
PUPPET_STATUS=$(service puppet status)
echo "PUPPET_AGENT_STATUS : "$PUPPET_STATUS"
``` |
| **UnsignCertOfPuppetAgent** | ```bash
#!/bin/sh
MASTER_ENDPOINT=#CfgMgmtEndpoint#
sed -i '/search/d' /etc/resolv.conf
curl -k "$MASTER_ENDPOINT:8140/packages/current/install.bash"| sudo bash
CMD="export PATH=$PATH:/opt/puppetlabs/puppet/bin:/opt/puppetlabs/puppet/lib:/opt/puppetlabs/puppet/bin/gem"
CMD="service puppet start"
PUPPET_STATUS=$(service puppet status)
echo "PUPPET_AGENT_STATUS : "$PUPPET_STATUS"
``` |

3. Click **Submit** after creating each script template.

*Create a workload provider type for Puppet*

Create a workload provider type for Puppet.

Create a workload provider type information for the Puppet provider.

Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to **Manage > Config Management > Workload Config Provider Types**.

   The Workload Config Provider Type window appears.
2. Click New.
3. Fill in the form fields (as shown in the table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (<strong>Puppet Enterprise</strong>) for the workload config provider type.</td>
</tr>
<tr>
<td>Product Type</td>
<td>Select (<strong>Puppet Enterprise</strong>) from the list.</td>
</tr>
<tr>
<td>Config CI</td>
<td>Select (<strong>u_cmdb_ci_puppet_enterprise</strong>) from the list.</td>
</tr>
<tr>
<td>Credential Resolver</td>
<td>Select (<strong>PuppetEnterpriseCredentialResolver</strong>) from the list.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Server Type</td>
<td>Enter Enterprise, Opensource in this field.</td>
</tr>
<tr>
<td>Version</td>
<td>Enter the version number 1.0.</td>
</tr>
<tr>
<td>Credential Type</td>
<td>Enter the name (sn_puppet_config_puppet_enterprise_credentials) of the credentials table for this workload config provider type. The name of the credentials table can vary based on application.</td>
</tr>
</tbody>
</table>

4. In the Workload Provider Properties section, insert a new row with the following values:

<table>
<thead>
<tr>
<th>Key</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environment</td>
<td>Test</td>
<td>Production</td>
</tr>
</tbody>
</table>

5. Click Submit.

6. Open the Puppet Enterprise record. In the Provider Cloud Template Mappings section, update the Cloud Script Template field (for the BootstrapNode provider operation) with the value InstallPuppetAgent.
7. Click **Update**.

*Create a MID Server capability for a Puppet provider.*

Create a MID Server capability for a Puppet provider
Create a capability for the Puppet provider in your MID Server and associate this capability with a MID Server. A capability is required for each MID Server to work with the Cloud Management application.

Role required: sn_cmp.cmp_root_admin.

1. Navigate to MID Server > Capabilities.
2. Click New to create a capability with the name Puppet.
3. Click Submit.
   - Associate the Puppet capability with the MID Server.
5. Open your MID Server record and in the Related Links, click the Capabilities sub tab.
6. Click New.
7. Select Puppet from the Capability list and click Submit.

Create Puppet credentials for a Cloud Management scoped app.

Create Puppet credentials for a Cloud Management scoped app
Create credentials for the Puppet provider you created in this Cloud Management application example.

Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to Manage > Credentials.
2. Click New and select Puppet Enterprise Credentials.
   - The Puppet Enterprise Credentials window appears.
3. Click the Additional actions icon and navigate to **Configure > Form Layout**.
4. Select the following attributes from the **Available** column, move them to the **Selected** column, and click **Save**.
   - Name
   - Active
   - MID Servers
   - Order
   - Username
   - Password
   - Machine Username
   - Machine Password

5. Fill in valid values in the form fields in the Puppet Enterprise Credentials window and click **Submit**.

Create a Puppet configuration management provider
Create a Puppet configuration management provider
Create a Puppet configuration management provider and run Discovery on the Puppet provider to discover its resources.
Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to Manage > Config Management.
2. Click New.
3. Fill out valid values in the form fields and click Submit.
   The Puppet configuration management provider is created.
4. In the Cloud Admin Portal, navigate to Manage > Config Management.
5. Open the Puppet configuration management record and click Discover Now.
   The discovered resources for Puppet appear under Entities.

Create a blueprint for a Puppet provider

Create a blueprint for a Puppet provider

Create a blueprint to provision virtual machine instances for the Puppet provider.

Role required: sn_cmp.cmp_root_admin

1. In the Cloud Admin Portal, navigate to Design > Blueprint.
2. Click New.
3. Give the blueprint a name and select a category for the blueprint and click Save.
   The blueprint is created and appears in the Deployment Model tab.
4. Add the resources, vSphere Datacenter and virtual server, to the blueprint.
5. Create associations to the Blueprint container resource to virtual server and the virtual server to the vSphere Datacenter.
6. Click Logical Datacenter and add the following constraints:
   - vSphere Datacenter
   - AWS Datacenter
7. Click the Operations tab and select Provision to display steps associated with the blueprint.
   The attributes appear in the right side of the window.
9. Select the attributes WorkloadConfigProvider and WorkloadConfigProvider Type and click Save.
10. Click New to add the BootstrapNode resource operation and update the following operation attributes with their respective mappings.
    - ScriptParameters:
      - "WorkloadConfigProviderType":"${parameter.WorkloadConfigProviderType}"
      - CfgMgmtEndpoint:
        - "${ci.cmdb_ci_workload_config_provider[${parameter.WorkloadConfigProvider}].url}"
11. Click Save.
12. Click New to add RegisterNode and ExecuteConfigpackages resource operations one by one.
13. After the three resource operations are added, click Publish to Catalog.

Publish the Puppet scoped application

Publish the Puppet scoped application

After you create the cloud API scoped application for the Puppet provider, validate the scoped application and then publish it to the ServiceNow Store.

You have to meet certain prerequisites before you can publish to the ServiceNow Store. See Publish an application to the ServiceNow Store.

Once you publish the scoped application to the ServiceNow Store, the application can be imported and is ready to be used by everyone.
Create a Google Cloud Platform scoped application for Cloud Management

This example shows you how to create a scoped application for Google Cloud Platform with the Cloud Management application.

- Role required: sn_cmp.cmp_root_admin
- Ensure that you have met all the prerequisites to publish applications in the ServiceNow Store. See the ServiceNow Store documentation for the required details.
- Creation of a repository in a version control system like Github
- an active google compute account
- Successful MID Server communication with Google compute cloud.

You can publish this scoped application to the ServiceNow Store. A Linux virtual machine is used in this example.

Sequentially perform the following tasks to successfully create a Google Cloud Platform scoped application.

1. Create an application in the Studio.
2. Switch to the newly created application to set dependencies and create application settings for the new application.
3. Create a table for storing credential information to access the Google Cloud Platform server.
4. Create a CI class to manage all your Google Cloud Platform instances.
5. Create a capability for the Google Cloud Platform provider in MID Server. A capability is required for each MID Server to work with the Cloud Management application.
6. Create script includes to be used by the cloud APIs to communicate with the Google Cloud Platform server.
7. Configure MID Server jar files.
8. Map the Google Cloud Platform provider to credentials.
9. Create a cloud API provider for Google Cloud Platform.
10. Create cloud APIs.
11. Create a resource block.
12. Add Google datacenter as a host to the Virtual Server resource block.
13. Create a blueprint.
14. Test the scoped application.
15. Publish the scoped application to the ServiceNow Store.

Create an application for Google Cloud Platform in Studio

Create all your Google Cloud Platform configuration information in the form of an application using the Studio.

Role required: sn_cmp.cmp_root_admin

1. Navigate to System Applications > Studio.
   The Load Application pop-up window appears.
2. Click Create Application.
   The Create Application dialog box appears.
3. Click **Create** in the **Create Custom Application** section.
4. Fill in the form fields (as shown in the table).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a name (GCP demo) for the application.</td>
</tr>
<tr>
<td>Scope</td>
<td>The scope of the application, (sn_gcp_demo*), automatically appears.</td>
</tr>
</tbody>
</table>

5. Click **Create**.
   A confirmation dialog box appears.

6. Click **OK**.
   An application creation pop-up window appears on the successful creation of the application.

**Set dependencies on Google Cloud Platform application**
Set dependencies on Google Cloud Platform application
Create application settings and set dependencies to associate Cloud Management plugins with the Google Cloud Platform scoped application.

Role required: sn_cmp.cmp_root_admin

1. Click **System Applications > Studio**.
2. From the Applications list, click the application you created, **GCP demo**.
The Welcome to Studio home page appears.

3. Navigate to **File > Settings**.
   The Application Settings window appears.
### Custom Application

**GCP Credentials**

<table>
<thead>
<tr>
<th>Name</th>
<th>GCP Credentials</th>
<th>Version</th>
<th>1.0.0</th>
</tr>
</thead>
</table>

**Application Scoping**

Application scope will provide better protection for any system installing this application. Leave empty to create your application in the global scope. Enable Application administration to prevent System Administrators from accessing the application.

<table>
<thead>
<tr>
<th>Scope</th>
<th><code>x_snca_grp_credential</code></th>
</tr>
</thead>
</table>

**Design and Runtime**

Changing these options may affect how your application runs once it's been installed on a target instance.

<table>
<thead>
<tr>
<th>JavaScript Mode</th>
<th>ES5 Standards Mode</th>
<th>Runtime Access Tracking</th>
<th>Tracking</th>
<th>Restrict Table Choices</th>
</tr>
</thead>
</table>

**Subscription Management**

If this application is licensable, set the subscription requirement and model.

<table>
<thead>
<tr>
<th>Licensable</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscription requirement</td>
<td>Monitor</td>
</tr>
<tr>
<td>Subscription Model</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

**Primary menu**

Enter a new or choose an existing name to define the primary menu in the navigator for this application. Any module created for this app will reside by default in this menu.

<table>
<thead>
<tr>
<th>Menu</th>
<th>GCP Credentials</th>
</tr>
</thead>
</table>
4. In the Runtime Access Tracking list, select None. The default value is Tracking.
5. Click Update.
6. In Related Lists, click the Dependencies tab.
7. Click Edit. Select the following plugins from the Collection list and move them to the Dependencies list, to associate the plugins with the Google Cloud Platform scoped application.
   - Cloud API
   - Cloud Management
   - Cloud Config Management
   - Cloud Management Core
8. Click Save.
   All the plugins, added as dependencies, appear under the Dependencies tab.

Create table for storing Google Cloud Platform credential information
Create a table for storing credential information pertaining to the Google Cloud Platform provider. This information is used to connect to the Google Cloud Platform server.

Role required: sn_cmp.cmp_root_admin.
1. Navigate to System Definition > Tables.
2. Click New.
3. Fill in the form fields (as shown in the table below).
Field | Description
--- | ---
Label | Enter a unique label for the table (GCP Credentials). The label appears on list and form views for the table.
Name | The table name is automatically populated based on the table label and a prefix. Ensure the table name reads `sn_gcp_demo_gcp_credentials`. For a table in a scoped application, the name is prefixed with a namespace identifier to indicate that it is part of an application.
Extends table | Select the table to extend (Credentials). Extending a base table incorporates all the fields of the original table and creates system fields for the new table. You can extend tables that are marked as extensible if they are in the same scope or if they allow configuration from other scopes.

4. Right-click the header and select Save. The GCP credentials table appears.
5. Click Application Access to define the scope protection for the table.
6. Select the following check-boxes:
   - Can Create
   - Can Update
   - Can Delete
7. Right-click the header and select Save. You need to add information for the PrivateKey column.
8. Click Columns and then click New (next to Table Columns).
9. Fill in the form fields (as shown in the table below):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Select String from the list.</td>
</tr>
<tr>
<td>Column Label</td>
<td>Enter a label (PrivateKey) for the column.</td>
</tr>
<tr>
<td>Column Name</td>
<td>Enter a name (private_key) for the column.</td>
</tr>
<tr>
<td>Max Length</td>
<td>Enter the value 4000.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this check box.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Select this check box.</td>
</tr>
</tbody>
</table>

10. Click Submit. You now need to add information for the Client Email column.
11. Click New.
12. Fill in the form fields (as shown in the table below):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Select String from the list.</td>
</tr>
<tr>
<td>Column Label</td>
<td>Enter a label (Client Email) for the column.</td>
</tr>
<tr>
<td>Column Name</td>
<td>Enter a name (Client_email) for the column.</td>
</tr>
</tbody>
</table>
13. Click **Submit**.
    You now need to add information for the **Project Name** column.

14. Click **New**.
15. Fill in the form fields (as shown in the table below):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Length</td>
<td>Enter the value <strong>100</strong>.</td>
</tr>
<tr>
<td>Active</td>
<td>Select this check box.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Select this check box.</td>
</tr>
</tbody>
</table>

16. Click **Submit**.
17. Click the **User Menu** list and select **Elevate Roles**.
18. Select the **security_admin** check box and click OK.
19. In the **GCP Credentials** table, scroll down to **Related Links > Access Controls** and open the record with the read operation.
20. Select ***** in the list next to the **Name** list to allow access to records in the Credentials table.
21. In the **Requires role** section, add a role with the value, **mid_server**, to the Access Control record.
22. Click **Save** and then click **Update**.
    The Verify Security Rules dialog box appears.
23. Click **Continue**.
24. Scroll down in the table to Related Links and click **Show List**.
25. Click **New**, next to GCP Credentials.
26. Click the **Additional actions** icon and then select **Configure > Form Layout**.
    The Configuring GCP Credentials form window appears.
27. Select the following from the Available column and move them to the Selected column.
    - PrivateKey
    - ProjectName
    - ClientEmail
28. Click **Save**.

**Create CI class provider for Google Cloud Platform**
Create CI class provider for Google Cloud Platform
Create a CI class (table) to manage all your Google Cloud Platform instances.
Role required: **sn_cmp.cmp_root_admin**

1. Navigate to **CI Class Manager**.
2. Click **Hierarchy** and search for **Logical Datacenter**.

3. Click the Open Menu icon (next to **Logical Datacenter**) and select **Add Child Class** to create a new child class.

4. Enter a name (**Google Datacenter**) for the class in the **Display Name** field.

5. Keep clicking **Next** till you reach **Done**.
6. Click **Done** to create the Google Datacenter.

**Create a capability in MID Server for Google Cloud Platform**
Create a capability in MID Server for Google Cloud Platform
Create a capability for the Google Cloud Platform provider in MID Server and associate this capability with a MID Server. A capability is required for each MID Server to work with the Cloud Management application.

Role required: sn_cmp.cmp_rootAdmin.

1. Navigate to **MID Server > Capabilities**.
2. Click **New** to create a capability with the name **Google**.
3. Click **Submit**.
   - You need to associate the **Google** capability with a MID Server.
4. Navigate to **MID Server > Servers**.
5. In the Related Links, click the **Capabilities** sub tab.
6. Click **New**.
7. Select **Google** from the **Capability** list and click **Submit**.

**Create script includes for Google Cloud Platform**
Create script includes for Google Cloud Platform
Create script includes to be used by the cloud APIs to communicate with the Google Cloud Platform server.

Role required: sn_cmp.cmp_rootAdmin

1. Navigate to **Mid Server > Script Includes**.
2. Click **New**.
3. Fill in the form details for each script include and click **Submit**. See **Script includes**.
   - You need to create two script includes (GoogleAPIInvoker and GoogleComputeVirtualMachine) for the cloud APIs to communicate with the Google Cloud Platform server. See **Sample script includes for Google Cloud Platform**.

**Create a MID Server JAR file for Google Cloud Platform**
Sample script includes for Google Cloud Platform
Several sample scripts are provided, by default, by the Cloud Management application. You need these script includes for the cloud APIs to communicate with the Google Cloud Provider server.

**GoogleAPIInvoker**

```javascript
var GoogleAPIInvoker = Class.create();

GoogleAPIInvoker.prototype = Object.extendsObject( CloudRESTAPIInvoker, { 
  initialize: function(url, scope, clientEmail, privateKey) { 
    CloudRESTAPIInvoker.prototype.initialize.call(this);
    this.endpointUrl = url;
    this.scope = scope;
    this.clientEmail = clientEmail;
    this.privateKey = privateKey;
  },

  setHeaders: function(headers) {
    this.headers = headers;
  },

  /* Other methods */

} )
```

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
setParameters: function(parameters) {
  this.parameters = parameters;
},

getEndpointUrl: function() {
  return this.endpointUrl;
},

ggetToken: function() {
  var tokenGenerator = new Packages.com.snc.cmp.connector.util.GoogleJwsUtil();
  this.info("GoogleAPIInvoker::getToken::scope  = " + this.scope + "
clientEmail = " + this.clientEmail + " privateKey  = " + this.privateKey);
  var accessToken = tokenGenerator.getToken(this.scope, this.clientEmail, this.privateKey);
  this.info("obtained accessToken : " + accessToken);
  return accessToken;
},

getHeaders: function(headers) {
  headers["User-Agent"] = "ServiceNow-Jakarta";
  headers["Content-Type"] = "application/json";
  headers["Authorization"] = "Bearer " + this.getToken();

  if (!JSUtil.nil(this.headers)) {
    for (var header in this.headers) {
      parameters[header] = this.headers[header];
    }
  }
},

getParameters: function(parameters) {
  if (!JSUtil.nil(this.parameters)) {
    for (var param in this.parameters) {
      parameters[param] = this.parameters[param];
    }
  }
},

type: "GoogleAPIInvoker"
});

GoogleComputeVirtualMachine

var GoogleComputeVirtualMachine = Class.create();

GoogleComputeVirtualMachine.prototype = Object.extendsObject(CloudAPIBase, {
  CloudComputeDTO: Packages.com.snc.cmp.connector.dto.CloudCompute,
  scope: "https://www.googleapis.com/auth/cloud-platform",

  initialize: function(parameters, headers) {
    CloudAPIBase.prototype.initialize.call(this);

    this.parameters = parameters;
    this.headers = headers;
    this.clientEmail = parameters.get('ClientEmail');
    this.privateKey = parameters.get('PrivateKey');
  },

  createVirtualMachine: function(projectId, location, instanceRequest, params) {

var jsonObj = new JSON();

// send the request to GCP
var computeEndpoint = "https://www.googleapis.com/compute/v1/projects/" + projectId + "/zones/" + location + "/instances";
var apiInvoker = new GoogleAPIInvoker(computeEndpoint, this.scope, this.clientEmail, this.privateKey);

var responseStr = apiInvoker.invokePost(jsonObj.encode(instanceRequest));
var responseObj = apiInvoker.getResponse(responseStr);

// sleep for a few seconds to let the vm initialize
Packages.java.lang.Thread.sleep(20000);

// fetch the details of the vm
computeEndpoint = "https://www.googleapis.com/compute/v1/projects/" + projectId + "/zones/" + location + "/instances/" + instanceRequest.name;
apiInvoker = new GoogleAPIInvoker(computeEndpoint, this.scope, this.clientEmail, this.privateKey);
responseObj = new JSON().decode(apiInvoker.invokeGet());
var computeNode = new this.CloudComputeDTO();

// setup the node properties
computeNode.setLocation(location);
computeNode.setNodeName(instanceRequest.name);
computeNode.setNodeId(responseObj.id);
computeNode.setState(responseObj.status);
computeNode.setHardwareId(params.hardwareId);
computeNode.setImageId(params.imageId);

return computeNode;
}


Create a MID Server JAR file for Google Cloud Platform

MID Server JAR Files allow you to upload a JAR file to an instance and synchronize it to all MID Servers. The JAR file provides REST APIs to communicate with Google cloud.

- Role required: sn_cmp.cmp_root_admin
- You should have downloaded the JAR file, gcp-util-1.0.jar, onto your local system.

1. Navigate to **MID Server > JAR Files**.
2. Click **New**.
3. Enter a name (**JSON Web Signature Util**) and a version (**1.0**) in the **Name** and **Version** fields.
4. Click the Manage Attachments icon to attach the JAR file, **gcp-util-1.0.jar**.
5. Click **Update**.

Create a cloud API provider for Google Cloud Platform

Create a cloud API provider for Google Cloud Platform

Role required: sn_cmp.cmp_root_admin

1. In the Cloud Admin Portal, navigate to **Design > Cloud API**.
2. Click **New**.
   - The Add Cloud Provider dialog box appears.
3. Fill in the form details (as shown in the table below).
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Provider</td>
<td>Enter the name (Google) of the cloud provider.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a short description for the cloud provider.</td>
</tr>
<tr>
<td>Datacenter class</td>
<td>Select a datacenter class (Google Datacenter) for the provider.</td>
</tr>
</tbody>
</table>

4. Click Submit.  
The Cloud Provider details appear.

5. Click New next to Cloud Products.

6. Fill in the form fields (as shown in the table below).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (Google Compute) for the cloud provider.</td>
</tr>
<tr>
<td>Service Category</td>
<td>Select a service category (Compute) for the cloud provider.</td>
</tr>
</tbody>
</table>

7. Click Submit.

8. Open the Google Compute record.

9. Click New, next to Cloud Resources Types.

10. Fill in the form fields (as shown in the table below).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (Compute.Engine/VM Instances) for the resource type.</td>
</tr>
<tr>
<td>CI Class</td>
<td>Enter a CI class (cmdb_ci_vm_instance) for the resource type.</td>
</tr>
</tbody>
</table>

11. Click Submit.

---

**Create a cloud API for Google Cloud Platform**

Create a cloud API for Google Cloud Platform
Create a cloud API for the Google Cloud Platform provider.

Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to Design > Cloud API.
2. Click API and then click New.
   The Add Cloud API dialog box appears.
3. Fill in the form details (as shown in the table below).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud API</td>
<td>Enter a name (Google Compute Platform API) for the cloud API.</td>
</tr>
<tr>
<td>Interface</td>
<td>Select an interface (Compute Interface) for the cloud API.</td>
</tr>
<tr>
<td>Product</td>
<td>Select a product (Google Compute) for the cloud API.</td>
</tr>
</tbody>
</table>
4. Click **Submit**. The Google Compute Platform API information appears.

5. In the CAPI Method Mappers tab, search for **CreateNode** and open the **CreateNode** record.

6. In the **Request Script** field, click the information icon to open the record.

7. Fill in the form fields (as shown in the table below).

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (google-compute-1.0-CreateNode) for the script.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description (Script for CAPI - Google Compute Platform API1.CreateNode) for the script.</td>
</tr>
</tbody>
</table>
Script

function createVirtualMachine() {
    var jsonObj = new JSON();

    var location = this.parameters.get('Location');
    var vmName = "" + this.parameters.get('NodeName');
    var computeProfile = this.parameters.get('ComputeProfile');
    var osProfile = this.parameters.get('OsProfile');

    var mappedHardware = jsonObj.decode(computeProfile);
    var hardwareId = mappedHardware.HardwareId;
    var localStorage = mappedHardware.LocalStorage;

    var mappedOsProfile = jsonObj.decode(osProfile);
    var imageId = mappedOsProfile.ImageId;

    var projectName = this.parameters.get('Project');
    var gcpvm = new GoogleComputeVirtualMachine(this.parameters,
    this.headers);

    ms.log('google-compute-1.0-
CreateNode::createVirtualMachine
  + 'location = ' + location + '
  + vmName = ' + vmName + '
  + computeProfile = ' + computeProfile + '
  + osProfile = ' + osProfile + '
  + mappedHardware = ' + mappedHardware + '
  + hardwareId = ' + hardwareId + '
  + localStorage = ' + localStorage + '
  + mappedOsProfile = ' + mappedOsProfile + '
  + imageId = ' + imageId + '
  + projectName = ' + projectName + '}

    // validate vm name
    if (vmName.endsWith("-"))
        vmName = vmName.substring(0, vmName.length - 1);

    vmName = vmName.substring(0, 64); // maximum 64 chars in

    var instanceRequest = {
        "name": vmName,
        "zone": "projects/" + projectName + "/zones/about",
        "machineType": "projects/" + projectName + "/zones/about/machineTypes/about",
        "disks": [
            {
                "type": "PERSISTENT",
                "boot": true,
                "mode": "READ_WRITE",
                "autoDelete": true,
                "deviceName": vmName,
                "initializeParams": {
                    "sourceImage": imageId,
                    "diskType": "projects/" + projectName + "/zones/about/diskTypes/pd-standard",
                    "diskSizeGb": localStorage
                }
            }
        ],
        "canIpForward": false,
        "networkInterfaces": [
            {
                "network": "projects/" + projectName + "/global/networks/default",
                "subnetwork": "projects/" + projectName + "/regions/about/subnetworks/default",
                "accessConfigs": [
                    {
                        "name": "External NAT",
                        "type": "ONE_TO_ONE_NAT"
                    }
                ]
            }
        ],
        "description": 
        "scheduling": {
            "preemptible": false,
            "onHostMaintenance": "MIGRATE",
            "automaticRestart": true
        }
    };

    var params = {
        "hardwareId": hardwareId,
        "imageId": imageId
    };

    try {
        vm = gcpvm.createVirtualMachine(projectName,
        location, instanceRequest,
        params);
    } catch (e) {
        ms.log(e);
        throw e;
    }

    return vm;
}
8. Click **Update**.
The CAPI Method Mapper window appears.
9. Click the back icon, next to CAPI Method Mapper, to get back to the Cloud API window.
10. Click the **API Config Overrides** sub tab and then click **New**.
The API Config Overrides window appears.
11. Click the search icon next to the **Config Parameter** list.
12. Click **New** to create a config parameter.
13. Fill in the form fields (as shown in the table below).
   
   You need to create three API config overrides; one for PrivateKey, one for ClientEmail, and one for Project.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Config Parameter Name</td>
<td>Enter a name for the config parameter.</td>
</tr>
<tr>
<td></td>
<td>• PrivateKey</td>
</tr>
<tr>
<td></td>
<td>• ClientEmail</td>
</tr>
<tr>
<td></td>
<td>• Project</td>
</tr>
<tr>
<td>Default value</td>
<td>Enter a default value for a config parameter.</td>
</tr>
<tr>
<td></td>
<td>• $(CloudCredential.private_key) for PrivateKey</td>
</tr>
<tr>
<td></td>
<td>• $(CloudCredential.client_email) for ClientEmail.</td>
</tr>
<tr>
<td></td>
<td>• $(CloudCredential.project_name) for Project</td>
</tr>
<tr>
<td>Config data type</td>
<td>Select a config data type from the list.</td>
</tr>
<tr>
<td></td>
<td>For all the three config parameters, (PrivateKey, ClientEmail, and Project), select <strong>String</strong>.</td>
</tr>
<tr>
<td>Config type</td>
<td>Select a config type from the list.</td>
</tr>
<tr>
<td></td>
<td>For all the three config parameters, (PrivateKey, ClientEmail, and Project), select <strong>Config</strong>.</td>
</tr>
</tbody>
</table>

14. Click **Update**.
The API Config Overrides window appears.
15. Enter a value in the **Override value** field. Based on the config parameter you created, enter the appropriate value.

<table>
<thead>
<tr>
<th>Config Parameter</th>
<th>Override value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PrivateKey</td>
<td>$(CloudCredential.private_key)</td>
</tr>
<tr>
<td>ClientEmail</td>
<td>$(CloudCredential.client_email)</td>
</tr>
<tr>
<td>Project</td>
<td>$(CloudCredential.project_name)</td>
</tr>
</tbody>
</table>

16. Click **Submit**.

**Create a resource block for Google Cloud Platform**
Create a resource block for Google Cloud Platform

Resource blocks are the building blocks of cloud catalog items. Create a resource block for the Google Cloud Platform provider, based on a CI type from the CMDB.
Role required: sn_cmp.cmp_root_admin

1. In the Cloud Admin Portal, navigate to Design > Resource Blocks.
2. Click New to open the Create Resource Block dialog box and then fill in the form. For a detailed description of each field, see Create a custom resource block.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (Google Datacenter) for the resource block.</td>
</tr>
<tr>
<td>Layer</td>
<td>Select the type of resource block (Logical Datacenter).</td>
</tr>
<tr>
<td>CI Type</td>
<td>Select the CI type (Google Datacenter) that the resource block represents.</td>
</tr>
<tr>
<td>Guest Interface</td>
<td>Select a default guest interface (Compute Interface) to use when you add a resource block to a blueprint.</td>
</tr>
</tbody>
</table>

3. Click Submit. The resource block information appears.
4. Click the Resource Script sub tab and then click New.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name (Create_VM_GCP_Response_Processor) for the resource script.</td>
</tr>
<tr>
<td>Type</td>
<td>Select a type (Response Processor) for the script.</td>
</tr>
</tbody>
</table>
Enter the following script:

```javascript
function processResponse(response, cloudServiceAccountId, ldc, correlationId, step, requestorContext) {
    var cloudModelString = [];
    var apiObj = global.JSON.parse(step);
    var vmResponse = global.JSON.parse(response);
    var reqContext = global.JSON.parse(requestorContext);
    var status_map = {'RUNNING': 'on',
                      'SUSPENDED': 'paused',
                      'STOP': 'off',
                      'TERMINATED': 'terminated',
                      'ERROR': 'error',
                      'UNRECOGNIZED': 'off',
                      'PENDING': ''
                     }
    if(Array.isArray(vmResponse))
        vmResponse = vmResponse[0];

    var vmInfo = {
        'cmdb_ci_vm_instance': {
            'validator': 'virtual_machine_create_update_validator',
            'identification': {
                'cmdb_ci_cloud_service_account': {
                    'criterion': {
                        'object_id': cloudServiceAccountId
                    }
                },
                'u_cmdb_ci_google_datacenter': {
                    'criterion': {
                        'object_id': apiObj.ldcObjectId
                    }
                }
            },
            'cmdb_ci_vm_instance': {
                'criterion': {
                    'object_id': vmResponse.nodeId
                }
            },
            'attributes': {
                'name': vmResponse.nodeName,
                'object_id': vmResponse.nodeId,
                'state': status_map[vmResponse.state],
                'ip_address': vmResponse.publicIps[0],
                'assigned_to': reqContext.userId,
                'assignment_group': reqContext.groupId
            }
        }
    }
    cloudModelString.push(vmInfo);
    return global.JSON.stringify(cloudModelString);
}
```
5. Click **Submit**.
6. Click the **Operations** sub tab and then click **Steps**.
7. Select **ConnectAndCreateVirtualMachine** from the **Operations** list.
8. Click the **Add a Step** plus icon to open the Add Operation Steps dialog box and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td>Select <strong>Invoke Cloud API</strong> from the list.</td>
</tr>
<tr>
<td>API Provider</td>
<td>Select <strong>Google Compute</strong> from the list.</td>
</tr>
<tr>
<td>API Interface</td>
<td>Select <strong>Compute Interface</strong> from the list.</td>
</tr>
<tr>
<td>API Method</td>
<td>Select <strong>CreateNode</strong> from the list.</td>
</tr>
<tr>
<td>CAPI Version</td>
<td>Enter the value <strong>1.0</strong> in the field.</td>
</tr>
</tbody>
</table>

9. Click **Submit**.
10. Click **Response Processor** and then click the plus icon. The Add Response Processor dialog box appears.
11. Select **Create_VM_GCP_Response_Processor** from the **Script Name** list and click **Submit**.
12. Click the **Draft** toggle button. The Google Datacenter resource block moves from the draft to the published state and no further changes can be made to the resource block. Changes to a resource block can be made only if the status is set to **Draft**.

**Configure a virtual server for Google Cloud Platform**

Configure a virtual server for Google Cloud Platform
Add the Google Datacenter as a host to the Virtual Server resource block.

Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to **Design > Resource Blocks**.
2. Search for the Virtual Server resource block and open the Virtual Server resource block record.
3. Click **Published** to move the resource block to draft mode.
4. Click **Host Resource** and then click **New**.
5. From the **GuestResource** list, select **Virtual Server**.
6. From the **HostResource** list, select **Google Datacenter**.
7. Click **Submit**.
8. Navigate to the Virtual Server resource block.
9. Click the **Draft** toggle button to move the resource block to a published state.

   Google Datacenter is listed as a host resource for Virtual Server in the Host Resource tab.

**Create credential mapping for Google Cloud Platform provider**

Create credential mapping for Google Cloud Platform provider
Map the Google Cloud Platform provider to the GCP credentials table. Each credential is mapped to a particular provider.

Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to **Manage > Credentials > Provider Credential Mapping**.
2. Click **New**.
3. Fill in the form fields (as shown in the table).
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider</td>
<td>Select the name of the provider (Google Datacenter) from the list.</td>
</tr>
<tr>
<td>Credential</td>
<td>Select the credentials (GCP Credentials) for the provider from the list.</td>
</tr>
<tr>
<td>Available</td>
<td>Select this check box. This check box indicates that the Google Cloud Platform provider appears in the list of providers in the Create Cloud Accounts dialog box.</td>
</tr>
<tr>
<td>URL Mandatory</td>
<td>Ensure that this check box is not selected.</td>
</tr>
<tr>
<td>Resource Block</td>
<td>Select the resource block to be used in discovering resources.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

Create a blueprint for Google Cloud Platform
Create a blueprint for Google Cloud Platform
Create a blueprint to provision virtual machine instances in the Google compute engine.

Role required: sn_cmp.cmp_root_admin.

1. In the Cloud Admin Portal, navigate to **Design > Blueprint**.
2. Click **New**, enter a name (**GCE VM Instance**), and then select a category (**Compute**).
3. Click **Submit**.
   The blueprint information appears.
4. Click the + icon next to Resources, search for Virtual Server and click **Virtual Server**.
   Virtual Server is added to the layout pane.
5. Search for Logical Datacenter in Resources and click **Logical Datacenter**.
   Logical Datacenter is added to the layout pane.
6. Connect the Blueprint container resource to the Virtual Server. To create a connection between resources, select the parent resource (Blueprint Container), click the connection pad in the resource, and then drop the connection in the Virtual Server with a dotted outline.
7. Connect the Virtual Server to the Logical Datacenter.
8. Click **Operations** and then click **Provision**.
9. Ensure that you select the check boxes for the following attributes:
   - Application
   - Business Service
   - Cloud Account
   - Cost Center
   - Lease End Date
   - Location
   - ScheduleProfile
   - ScheduleTimezone
   - StackName
   - UserGroup

10. Click **Virtual Server** and then click **Provision** to view Virtual Server attributes.
11. Select the check boxes for the following attributes:
   - ComputeProfile
- NodeName
- OSProfile
- SSHKey

12. Click **Publish to Catalog** to generate catalog forms.
13. Click **GCE VM Instance** under Request Operation.
   Operation inputs of GCE VM Instance blueprint appear.
14. Click **Form** and then click **Location**.
15. Scroll down to **Rules** section.
16. Click **TriggerChangeEvent Virtual_Server_OSProfile** to edit it.
17. On the **Actions** tab, click **New**, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name <em>(Reload from pool)</em> for the action.</td>
</tr>
<tr>
<td>Action Type</td>
<td>Select a type of action <em>(Reload)</em> from the list.</td>
</tr>
<tr>
<td>Datasource</td>
<td>Select a datasource <em>(Reload From pool)</em> from the list.</td>
</tr>
<tr>
<td>TargetField</td>
<td>Select a target field <em>(Virtual_Server_OSProfile)</em> for the list.</td>
</tr>
</tbody>
</table>

18. Scroll down and enter the following two keys and their values.

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CloudAccount</td>
<td>CloudAccount</td>
</tr>
<tr>
<td>Location</td>
<td>Location</td>
</tr>
</tbody>
</table>

19. Click **Submit**.
20. Click **Location** under **Form Parameters**.
21. In the **Rules** section, click **TriggerChangeEvent Virtual_Server_ComputeProfile** to edit it.
22. Click the **Actions** tab, click **New**, and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name <em>(Reload from pool)</em> for the action.</td>
</tr>
<tr>
<td>Action Type</td>
<td>Select a type of action <em>(Reload)</em> from the list.</td>
</tr>
<tr>
<td>Datasource</td>
<td>Select a datasource <em>(Reload From pool)</em> from the list.</td>
</tr>
<tr>
<td>TargetField</td>
<td>Select a target field <em>(Virtual_Server_ComputeProfile)</em> for the list.</td>
</tr>
</tbody>
</table>

23. Scroll down and enter two keys and their values. See step 18 for the keys and their values.
24. Click **Submit**.

The blueprint is created successfully and can provision a virtual machine in Google Datacenter.

*Publish the Google Cloud Platform scoped application*

Publish the Google Cloud Platform scoped application

After you create the cloud API scoped application for the Google Cloud Platform provider, validate the scoped application and then publish it to the ServiceNow Store.
Certain prerequisites need to be met before you can publish to the ServiceNow Store. See Publish an application to the ServiceNow Store.

After you test the scoped application and then publish the application to the ServiceNow Store, the application is ready to be used by everyone.

Create a Google Cloud Platform scoped application for demo purposes
Create a scoped application for Google Cloud Platform, push it to your local repository, and continue working on it. This application can be used for demo or testing purposes.

Role required: sn_cmp.cmp_root_admin.

1. Create an application in the ServiceNow Studio.
2. Create credentials to be used in provisioning a virtual machine instance in Google Cloud Platform.
3. Create a capability for Google Cloud Platform in the MID Server and associate this capability with a MID Server. A capability is required for each MID Server to work with the Cloud Management application.
4. Create a cloud account and a service account.
5. Configure a datacenter.
6. Create an OS template, a compute template, an OS profile and a compute profile.
7. Create capacity configurations for Google datacenter.
8. Create application files. Application files can be moved from one environment to another.
9. Create a blueprint for a scoped application and test the blueprint.
10. Test the newly created scoped application.
11. Push the newly created scoped application to your local repository.

Your scoped application is ready to be used for demo and testing purposes.

Moving Cloud Management content across environments
You can move the content in the Cloud Management application from one environment to the other. For example, develop an application in an environment and then move the application and all its dependencies to another environment for testing or production.

Content entities that can be moved
The Cloud Management application consists of code, resources, content, and configurations.
You can move the following content entities across environments:
- Blueprints
- Resource blocks
- Cloud APIs

Use the update sets to move the content entities. Update sets, which are XML files, group all the content into named sets and then moves these sets, as a unit, to other systems for testing or deployment. For information on how system update sets work, see System update sets.

Content entity dependencies
Each of the content entities has some direct dependencies that are automatically moved with the content to the new environment.

The direct dependencies for each of these content entities are:
<table>
<thead>
<tr>
<th>Content entity</th>
<th>Dependencies</th>
</tr>
</thead>
</table>
| Blueprint      | - Resource blocks  
|                | - Cloud APIs  
|                | - MID scripts  
|                | - Resource pools  
|                | - Policies  
|                | - Script includes  
|                | - Cloud scripts  |
| Resource blocks | - Resource pools  
|                | - Resource operation policies  
|                | - Script includes  
|                | - Cloud APIs  
|                | - Dependant resource blocks  
|                | - MID scripts  |
| Cloud API      | - MID scripts  |

**Move a blueprint from one environment to another**

Use update sets to move a blueprint and its dependencies from one environment to another. Update sets let you group a blueprint and its dependencies into a named set and then move them as a unit to other systems for testing or deployment. For example, you can move a blueprint from a development environment to a production environment.

Role required: sn_cmp.cloud_service_designer or admin

- Users with the sn_cmp.cloud_service_designer role can only export update sets.
- Users with the admin role can export and import update sets.

Package the blueprint as an update set. Then export the update set from its current environment and import it to another environment.

1. In the Cloud Admin Portal, navigate to **Design > Blueprints**.
   All blueprints, in published and draft mode, appear in the Blueprints window.
2. Export the blueprint.
   a) Click the Export icon (Leod) for the blueprint that you want to export.
   The Blueprint Exporter window appears.
<table>
<thead>
<tr>
<th>Type</th>
<th>Policies</th>
<th>Policy Name</th>
<th>Policy Trigger</th>
<th>Target Entity</th>
<th>Status</th>
<th>Updated by</th>
<th>Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD Server Script Includes</td>
<td></td>
<td>Scheduling Default</td>
<td>on Catalog Item launch</td>
<td>Published</td>
<td>system</td>
<td>2018-11-05 14:36:44</td>
<td></td>
</tr>
</tbody>
</table>
The Blueprint Exporter window has a list of all the indirect dependencies (Policy, MID Server Script Includes, MID Server Script Files, Script Includes, and Workflows) for the blueprint. Objects such as policies, pools, MID scripts, script includes, and workflows are not directly a part of a blueprint, but a blueprint might depend on these objects to work correctly.

b) In the Blueprint Exporter window, click an object in the Type column. In the right column, select the object's corresponding entries that you want to export with the blueprint. For example, if you select MID Server Script Includes, all the corresponding script includes appear in the right column.

If a workflow contains subworkflows, select the parent workflow and all the subworkflows in order for the workflow to be exported. If there are any custom activities associated with a workflow, export those activities first and then export the workflow.

c) Click **Next**.

A window opens with a summary of the indirect dependencies that you chose to include in the export update set.

d) Click **Export Update Set**.

The Blueprint Exporter window opens with the **Success** check box selected in green indicating that the blueprint has been successfully exported along with all its dependencies. By default, all the files listed under Exported Update Set(s) get downloaded onto your system automatically. However, if the files are not automatically downloaded, follow these steps.
Blueprint Exporter - blueprint1

✅ Success

**Exported Update Set(s):**
- blueprint1 2018-05-02 17:57:51
- blueprint1 2018-05-02 17:57:53

**Metadata:**
- blueprint1 2018-05-02 17:57:53_metadata

**Log:**
- blueprint1 2018-05-02 17:57:53_log
Note: The number of update sets created is based on the scope of records that are being exported. For example, the blueprint that you are exporting has 100 records in all: 60 records are in scope one, 20 records are in scope two, and the remaining 20 records are in scope three. In such a scenario, three update sets are created. One update set for each scope.

3. Make sure that the files are downloaded. If the files are not downloaded, complete these steps.
   a) Click the metadata file.
      The metadata file mentions the order in which the exported files should be imported. In this example, the file blueprint10 is the first file to be exported followed by the file blueprint11.
   b) Based on the order mentioned in the metadata file, click the appropriate file to open it. A window opens with a list of all the files contained in the update set.
   c) To download the first XML file (in this example, the file is blueprint10) onto your system, click Export to XML. Import all the exported files into another environment. Users with the admin role can import the files. For example, you may have created the export update set in a system that runs the development environment and want to import it into another system that runs the production environment.
   d) Open the other XML file (in this example, the file is blueprint11) and download that file onto your system, too.

4. Import the blueprint.
   a) In the environment and the new instance where you want to import the files, enter Retrieved Update Sets in the filter navigator and then press the Enter key.
   b) Click the Import Update Set from XML related link.
   c) In the Import XML window that appears, click Choose File, select the export file, and click Upload.

Note: If a blueprint that you exported from the source environment exists in the target environment, and both the blueprints have a different sys_id, when you import the blueprint, an error appears and the import process stops. Export a different blueprint from the source environment or delete the blueprint with the same name in the target environment. If a blueprint that you exported from the source environment exists in the target environment and both the blueprints have the same sys_id, then the blueprint in the target environment gets updated with the blueprint that you exported from the source environment.

5. To verify that the blueprint is imported into the new environment, go to your instance in the new environment and in the Cloud Admin Portal, navigate to Design > Blueprints. The blueprint you imported should appear in the listed blueprints.

Move a resource block from one environment to the other
Use update sets to move a resource block and its dependencies from one environment to another. Update sets let you group a resource block and its dependencies into a named set and then move them as a unit to other systems for testing or deployment. For example, you can move a resource block from a development environment to a production environment.

Role required: sn_cmp.cloud_service_designer or admin
• Users with the sn_cmp.cloud_service_designer role can only export update sets.
• Users with the admin role can export and import update sets.

Package the resource block as an update set. Then export the update set from its current environment and import it to the other environment.

**Note:** If a resource block that you export from the source environment exists in the target environment, when you import this resource block, it results in two resource blocks with the same name: one resource block that existed before the import and another resource block which you just imported. For example, if you export the AWS Datacenter resource block from the development environment and you already have a resource block with the same name in the production environment, then when you import the AWS Datacenter resource block to the production environment, two resource blocks with the same name, AWS Datacenter, reside in the production environment.

1. In the Cloud Admin Portal, navigate to **Design > Resource Blocks.**
   All resource blocks, in published and draft mode, appear in the Resource Blocks window.
2. Export the resource block.
   a) Click the Export Resource block icon (Export Resource block icon) for the resource block to export. The Resource Block Exporter window appears.
a) In the Resource Block Exporter window, click an object in the Type column. In the right column, select the corresponding entries for the object that you want to export with the
resource block. For example, if you select MID Server Script Includes, all the corresponding script includes appear in the right column.

If a workflow contains subworkflows, select the parent workflow and all the subworkflows in order for the workflow to be exported. If there are any custom activities associated with a workflow, export those activities first and then export the workflow.

b) Click **Next**.
A window opens with a summary of the indirect dependencies that you chose to include in the export update set.

c) Click **Export Update Set**.

The Resource Block Exporter window opens with the **Success** check box selected in green indicating that the resource block has been successfully exported along with all its dependencies. By default, all the files listed under Exported Update Set(s) get downloaded onto your system automatically.
Resource Block Exporter - Azure Datacenter

☑ Success

Exported Update Set(s):
  Azure Datacenter0 2018-05-03 20:09:36
  Azure Datacenter1 2018-05-03 20:09:38

Metadata:
  Azure Datacenter 2018-05-03 20:09:38_metadata

Log:
  Azure Datacenter 2018-05-03 20:09:38_log

[ResourceBlock Exporter LOG] 2018-05-03 20:09:36:Exporting resource script - Stop_Azure_Datacenter0
[ResourceBlock Exporter LOG] 2018-05-03 20:09:36:Exporting resource script - Start_Azure_Datacenter0
3. Make sure that the files are downloaded. If the files are not downloaded, complete these steps.
   a) Click the metadata file.
      The metadata file mentions the order in which the exported files should be imported. In this example, the file Azure Datacenter is the first file to be exported followed by the file Azure Datacenter.
   b) Based on the order mentioned in the metadata file, click the appropriate file to open it.
      A window opens with a list of all the files contained in the update set.
   c) To download the first XML file (in this example, the file is Azure Datacenter) onto your system, click Export to XML.
   d) Open the other XML file (in this example, the file is Azure Datacenter1), and download that file onto your system.
      You must import all the exported files into another environment. For example, you may have created the export update set in a system that runs the development environment and want to import it into another system that runs the production environment.

4. Import the resource block.
   a) In the environment and the new instance where you want to import the files, enter Retrieved Update Sets in the filter navigator and then press the Enter key.
   b) Click the Import Update Set from XML related link.
   c) In the Import XML window that appears, click Choose File, select the export file, and click Upload.

Note: If a resource block that you exported from the source environment exists in the target environment, and both the resource blocks have a different sys_id, when you import the resource block, an error appears and the import process stops. Export a different resource block from the source environment or delete the resource block with the same name in the target environment. If a resource block that you exported from the source environment exists in the target environment and both the resource blocks have the same sys_id, then the resource block in the target environment gets updated with the resource block that you exported from the source environment.

5. To verify that the resource block is imported into the new environment, go to your instance in the new environment and in the Cloud Admin Portal, navigate to Design > Resource Blocks.
   The resource block you imported should appear in the listed resource blocks.

Move a cloud API from one environment to the other
Use update sets to move a cloud API from one environment to another. Update sets let you group a cloud API and its dependencies into a named set and then move them as a unit to other systems for testing or deployment. For example, you can move a cloud API from a development environment to a production environment.

Role required: sn_cmp.cloud_service_designer or admin
- Users with the `sn_cmp.cloud_service_designer` role can only export data sets.
- Users with the `admin` role can export and import update sets.

Package the cloud API as an update set. Then export the update set from its current environment and import it to the other environment.

**Note:** If a cloud API that you export from the source environment exists in the target environment, when you import this cloud API, it results in two cloud APIs with the same name: one cloud API that existed before the import and another cloud API which you just imported. For example, if you export the Ansible Configuration API from the development environment and you already have a cloud API with the same name in the production environment, when you import the Ansible Configuration API to the production environment, two cloud APIs with the same name, Ansible Configuration API, reside in the production environment.

1. In the Cloud Admin Portal, navigate to **Design > Cloud API > API**. All cloud APIs appear in the Cloud API window.
2. Export the cloud API.

   a) Click the Export Cloud API icon (\[\]) for the cloud API to export.
   
   The Cloud API Exporter window appears.
b) In the Cloud API Exporter window, click an object in the Type column. In the right column, select the corresponding entries for the object that you want to export with the cloud API. For example, if you select MID Script, all the corresponding MID scripts appear in the right column.

c) Click Next.
   A window opens with a summary of the indirect dependencies of the cloud API that you chose to include in the export update set.

d) Click Export Update Set.
   The Cloud API Exporter window opens with the Success check box selected in green indicating that the cloud API has been successfully exported along with all its dependencies. By default, all the files listed under Exported Update Set(s) get downloaded onto your system automatically.
Cloud API Exporter - Ansible Tower Configuration API

✅ Success

Exported Update Set(s):
- Ansible Tower Configuration API0 2018-05-03 22:38:41
- Ansible Tower Configuration API1 2018-05-03 22:38:43

Metadata:

Log:

3. Make sure that the files are downloaded. If the files are not downloaded, complete these steps.
   a) Click the metadata file.
      The metadata file mentions the order in which the exported files should be imported. In this example, the file titled Ansible Tower Configuration API0 is the first file to be exported, followed by the file titled Ansible Tower Configuration API1.
   b) Based on the order mentioned in the metadata file, click the appropriate file to open it.
      A window opens with a list of all the files contained in the update set.
   c) To download the first XML file (in this example, the file is API0 2018-05-03) onto your system, click Export to XML.
   d) Open the other XML file (in this example, the file is API1 2018-05-03), and download that file onto your system.
      You must import all the exported files into another environment. For example, you may have created the export update set in a system that runs the development environment and want to import it into another system that runs the production environment.

4. Import the cloud API.
   a) In the environment and the new instance where you want to import the files, enter Retrieved Update Sets in the filter navigator and then press the Enter key.
   b) Click the Import Update Set from XML related link.
   c) In the Import XML window that appears, click Choose File, select the export file, and click Upload.
      The cloud API gets imported to your new environment.

5. To verify that the blueprint is imported into the new environment, go to your instance in the new environment and in the Cloud Admin Portal, navigate to Design > Cloud API > API. The cloud API you imported should appear in the list of cloud APIs.

Cloud API (CAPI)
The Cloud API (CAPI) enables you to integrate Cloud Management with cloud providers using REST APIs.

CAPI components
Integration with cloud providers is performed through REST calls, such as PUT, GET, POST, and DELETE. CAPI provides the framework for you to integrate a cloud vendors REST API so that your instance can communicate with the cloud provider to manage cloud resources.

CAPI contains these components:
Providers

Cloud providers are the clouds that you can connect to. By default, Cloud Management includes the most commonly used providers, such as AWS, Azure, and VMware. Each provider has many products, each of which provide resource types. Each resource type maps to a single CI type. For example, the AWS provider includes the AWS Elastic Compute Cloud product, which includes the AWS::EC2::Instance resource type. This resource type is one of the most common cloud resources that you can create. It maps directly to the Virtual Machine Instance (cmdb_ci_vm_instance) CI type, where virtual machines are saved in the CMDB.

Interfaces

Interfaces define the framework that the system needs to structure the REST calls that the cloud provider APIs expect. Interfaces define operations, also called methods, and the parameters that each method requires.

Interfaces are reusable. If you extend CAPI to include new products and APIs, you can use existing interfaces to make the same REST calls.
APIs

CAPI APIs are the core component of CAPI that ties a product and interface together. The APIs include the actual code that the system executes.

Each CAPI API includes these components:

- **CAPI Method Mappers** provide the methods that are mapped to the operations defined in the interface. From CAPI method mappers, you create MID Server script includes in JavaScript to tell the cloud provider exactly what to do. It is through the script includes that the connection to the cloud provider occurs.

  **Note:** If you create custom CAPI APIs, the system provides an empty script include for you to customize. You can also modify existing script includes on method mappers if necessary. However, many of the default APIs that come with the Cloud Management application do not use modifiable script includes. The connections are hard-coded in Java. You can still use these APIs in new resource blocks that you create, but you cannot modify the APIs.

- **API Config Overrides** contain the identity, such as a key, and credentials, such as the secret key, and other important parameters required by the cloud provider. These parameters help the cloud provider perform the operations in the CAPI Method Mappers related list. API Config Overrides are necessary because, when the system calls the cloud provider API via REST, credential data is not included. Resource blocks use the parameters and values that you define in the API Config Overrides to query the credential store. When your API executes, the attributes are made available to all method calls in your script includes.

  The overrides are scoped only for this API. The overrides do not override anything in other APIs.

Because you can define multiple versions of a CAPI API with slight variations, you can extend (without overwriting) an existing API while retaining desired functionality.

This image illustrates the components of CAPI that are used when you provision an Azure virtual machine using the default CAPI settings provided with Cloud Management:
In this example, the Microsoft.Compute product is contained in the Azure provider. Azure uses the Microsoft.Compute product for virtual machines. In your instance, the Microsoft.Compute product maps to the Microsoft.Compute/virtualMachines resource type, which is associated to the Virtual Machine Instance CI type in the CMDB.

The Compute interface contains definitions for methods like CreateNode, which defines how to create the actual virtual machine. Of the many parameters that CreateNode uses, Location captures the datacenter where the virtual machine resides.

The Azure Compute API pulls together the Microsoft.Compute product and the structure defined in the Compute interface. The implementation of the CreateNode method calls the azure-compute-1.0-CreateNode MID Server script include, which calls the AzureComputeVirtualMachine MID Server script include. The script includes make the actual calls to the Azure API. To access the Azure account, the SecretKey, ClientID, TenantID, and other methods are passed in Config Overrides.
How CAPI integrates with the instance

CAPI integrates these components in your instance:

Cloud Management resource blocks

A resource block represents a single cloud resource, such as a virtual server, virtual server storage, or a datacenter. You can also think of it as a CI type in the CMDB. You put many resource blocks together in a blueprint, which appears as a catalog item (also called a stack) to your users in the Cloud Catalog.

In the system, each resource block is like a container that references CAPI, and links responses from the cloud provider to a specific CI. Resource blocks use:

- Operational steps that call CAPI for each operation, such as the provision operation, and pass along necessary parameter values that the cloud provider needs to execute the operation.
- Response processors that process and parse the REST response from the provider and update records in the CMDB.

The CMDB

Each resource block is based on a CI type from the CMDB. For Cloud Management, all cloud-related CI types are based on the Virtual Machine Object CI class, which provides all the attributes that you need for all the cloud resources supported by default. If a CI type for a cloud resource does not exist in the base system, you must create a new CI class and add the necessary attributes.

If you do create a new CI class, you must also create:

- A CI class for each of the resources that are available to your users. All CI classes are based on the virtual machine object class.
- An identification rule that specifies an Object ID. Whenever components of Cloud Management refer to a specific cloud resource in the CMDB, they need the Object ID to find the correct cloud resource.
- A relationship rule that specifies how the CI class for the resource is related to other CI classes. For example, a virtual server CI must have a Hosted on::Hosts relationship with a datacenter CI. These relationship rules are necessary for CI uniqueness when processed by the Identification and Reconciliation Engine (IRE). The combination of the service account, the object ID of the resource, and the datacenter (or location) where the resource is located determines the uniqueness.

MID Server script include

Each operation in the CAPI API has a MID Server script include that you configure. The script include calls the JavaScript classes that are already in other script includes in the system, or JavaScript classes that you create. Eventually, the invoker class is called to trigger the REST call. MID Server script includes are configured on your ServiceNow instance, but run on the MID Server.

This image illustrates how the components work together when a user provisions a resource from the Cloud User Portal:
REST calls to the cloud provider

REST calls to the cloud provider are triggered from the MID Server script includes that are referenced from scripted CAPI Method Mappers inside CAPI API records. To create your own CAPI APIs, or to create custom MID Server script includes (which are part of CAPI Method Mappers), you must understand:

- The classes that are available for you to call in MID Server script includes. See [CAPI classes in MID Server script includes](#).
- How the cloud provider implements REST. See:
  - [Azure REST API Reference](#)
  - [AWS Guides and API References](#)

For example, to learn how to make a REST call to Azure to create a resource group, review this Azure topic: [Resource Groups - Create Or Update](#). You can find the endpoint, parameters, and request body that Azure requires, and the responses it provides. You can see that:

- The endpoint is `management.azure.com`
- The method to call with a PUT operation is `subscriptions/{subscriptionId}/resourcegroups/{resourceGroupName}?api-version=2018-02-01`, where you specify the subscription ID, the resource group name, and the API version.

Remember that the REST API calls take place inside the MID Server script includes that are associated with CAPI API method mappers. Call the methods that CAPI already makes available for you by using the classes extended from `CloudAPIBase` and `CloudRESTAPIInvoker`. You can also create more script includes to extend these base classes and create your own classes. Familiarize yourself with these base classes and the methods available within them.

Get started here

1. [Review CAPI APIs that are provided with Cloud Management by default](#).
2. [Review the CAPI classes that are provided by default](#). These classes can be called from the MID Server script includes in your CAPI API operations.
3. Walk through the provisioning of an [Azure virtual machine](#) and an [AWS virtual machine](#) to see how the components work together. The Azure walkthrough uses a MID Server script include so you can see the various CAPI classes used in the provision operation. The AWS walkthrough does not use a MID Server script include.
4. [Add a product to an existing provider in CAPI](#).
5. [Create a CI class for a virtual cloud resource](#).
6. [Create or extend a CAPI interface](#).
7. [Create a CAPI API](#), and a custom MID Server script include that makes the REST calls to the cloud provider. An empty MID Server script include is always generated for new CAPI APIs. Modify it with the calls to other JavaScript classes and methods, such as the methods in the Invoker class.

Default CAPI APIs

Several CAPI APIs are provided by default with the Cloud Management application.
CAPI APIs refer to the API records that link a CAPI product and a CAPI interface. They do not refer to the CAPI classes that you can call from MID Server script includes in CAPI APIs. For an overview of CAPI classes, see CAPI classes in MID Server script includes.

View Cloud APIs by navigating to Cloud API > Library in the base system instance interface, or by navigating to Design > Cloud API in the Cloud Admin Portal.

<table>
<thead>
<tr>
<th>API</th>
<th>Interface</th>
<th>Product</th>
<th>Scripted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ansible Tower</td>
<td>Configuration Management</td>
<td>Ansible-Tower</td>
<td>YES*</td>
</tr>
<tr>
<td>Auto Scaling API</td>
<td>Auto Scaling Interface</td>
<td>AWS Elastic Compute Cloud</td>
<td>YES*</td>
</tr>
<tr>
<td>AWS Blob Storage API</td>
<td>Blob Storage Interface</td>
<td>AWS Simple Storage Service</td>
<td>NO</td>
</tr>
<tr>
<td>AWS Block Storage API</td>
<td>Block Storage Interface</td>
<td>AWS Elastic Block Storage</td>
<td>NO</td>
</tr>
<tr>
<td>AWS Compute API</td>
<td>Compute Interface</td>
<td>AWS Elastic Compute Cloud</td>
<td>NO</td>
</tr>
<tr>
<td>AWS Compute API</td>
<td>Compute Interface</td>
<td>AWS Elastic Compute Cloud</td>
<td>YES*</td>
</tr>
<tr>
<td>AWS Load Balancer API</td>
<td>Load Balancer Interface</td>
<td>AWS Elastic Load Balancing</td>
<td>NO</td>
</tr>
<tr>
<td>AWS Network API</td>
<td>Network Interface</td>
<td>AWS Virtual Private Cloud</td>
<td>NO</td>
</tr>
<tr>
<td>API</td>
<td>Interface</td>
<td>Product</td>
<td>Scripted</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>------------------------</td>
<td>----------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>AWS Stack Orchestration API</td>
<td>Stack Orchestration</td>
<td>AWS CloudFormation</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Alert API</td>
<td>Event Interface</td>
<td>Azure Alert</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Blob Storage API</td>
<td>Blob Storage Interface</td>
<td>Microsoft.Storage</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Block Storage API</td>
<td>Block Storage Interface</td>
<td>Microsoft.BlockStorage</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Compute API</td>
<td>Compute Interface</td>
<td>Microsoft.Compute</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Extensions API</td>
<td>Azure Extensions</td>
<td>Microsoft.Compute</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Load Balancer API</td>
<td>Load Balancer Interface</td>
<td>Microsoft.LoadBalancer</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Network API</td>
<td>Network Interface</td>
<td>Microsoft.Network</td>
<td>YES*</td>
</tr>
<tr>
<td>Azure Stack Orchestration API</td>
<td>Stack Orchestration</td>
<td>Azure ARM</td>
<td>YES*</td>
</tr>
<tr>
<td>Chef Server Configuration API</td>
<td>Configuration</td>
<td>Chef Server</td>
<td>NO</td>
</tr>
<tr>
<td>Infoblox IPAM API</td>
<td>IPAM Interface</td>
<td>Infoblox</td>
<td>NO</td>
</tr>
<tr>
<td>NetApp Block Storage API</td>
<td>Block Storage Interface</td>
<td>NetApp Storage</td>
<td>NO</td>
</tr>
<tr>
<td>NetApp Storage Server API</td>
<td>Storage Server</td>
<td>NetApp Storage</td>
<td>NO</td>
</tr>
<tr>
<td>Node Access API</td>
<td>Node Access Interface</td>
<td>NodeAccess</td>
<td>NO</td>
</tr>
<tr>
<td>OpenStack Blob Storage API</td>
<td>Blob Storage Interface</td>
<td>OpenStack Swift</td>
<td>NO</td>
</tr>
<tr>
<td>OpenStack Block Storage API</td>
<td>Block Storage Interface</td>
<td>OpenStack Cinder</td>
<td>NO</td>
</tr>
<tr>
<td>OpenStack Compute API</td>
<td>Compute Interface</td>
<td>OpenStack Nova</td>
<td>NO</td>
</tr>
<tr>
<td>OpenStack Network API</td>
<td>Network Interface</td>
<td>OpenStack-Neutron</td>
<td>NO</td>
</tr>
<tr>
<td>ServiceNow Store IPAM API</td>
<td>IPAM Interface</td>
<td>ServiceNow</td>
<td>YES*</td>
</tr>
<tr>
<td>SSH API</td>
<td>SSH Interface</td>
<td>SSH</td>
<td>NO</td>
</tr>
<tr>
<td>vSphere Block Storage API</td>
<td>Block Storage Interface</td>
<td>VMware VDisk</td>
<td>NO</td>
</tr>
<tr>
<td>vSphere Compute API</td>
<td>Compute Interface</td>
<td>VMware vSphere</td>
<td>NO</td>
</tr>
</tbody>
</table>

*You can customize how scripted APIs make REST calls to the cloud provider.

**CAPI classes in MID Server script includes**

Cloud Management provides several JavaScript classes that make REST calls to cloud providers. These classes are called from MID Server script includes (for Azure by default) or from Java calls in the system (for AWS by default). If you create custom providers, interfaces, or CAPI APIs (for any cloud provider), you can use MID Server script includes to call the classes.
**Note:** In CAPI APIs, there are no MID Server script includes (in Method Mappers) that you can customize to call the AWS classes. By default, the CAPI APIs for AWS make Java calls to then call the AWS classes. However, if you create CAPI APIs, you can also create custom script includes to call the classes.

View CAPI classes by navigating to **MID Servers > Script Includes** in the base system instance interface (not the Cloud Admin Portal). You search for the name of a class in the **Go to** search box using the **Name** column:

---

**Base classes**

These base classes are available by default. They form a basis for the Azure and AWS classes that make the actual REST calls.

**CloudAPIBase**

CloudAPIBase is a provider-agnostic API that communicates with all types of cloud APIs. It provides functions for basic debugging and logging, and for collecting proxy information on the MID Server. It also defines Headers, Parameters, Endpoints, and this.parameter.

**AWSCloudAPIBase**

This class is extended from CloudAPIBase and provides functions that call the invoker to make GET, PUT, POST, and DELETE requests. The class provides the option of an executeAction method, which is an AWS implementation of GET, or the invoke method, which provides REST API calls.

**AzureCloudAPIBase**

This class is extended from CloudAPIBase and provides functions that do these actions:

- Fetch resource IDs.
- Fetch endpoint properties.
- List resources.
- Call the invoker to make GET, PUT, POST, and DELETE requests.

AzureCloudAPIBase utilizes the _invokeRequest method to make REST API calls to Azure using the standard REST API calls.
Invoker classes

Invoker classes perform the actual REST calls to the cloud provider. If you create a custom CAPI interface and API, you should extend `CloudRestAPIInvoker` to create your REST calls.

CloudRestAPIInvoker

This class is an abstracted class for making REST API calls. If there are headers and/or parameters to send with a request, classes that implement the invoker must extend this object. They also must provide overrides for `getEndpointUrl` to return the targeted endpoint, and for `getHeaders` and `getParameters`. For example, with Azure REST APIs, authentication requires the passing of a token as a header. The extension implementing it should provide a response from `getHeaders` that performs the token creation or retrieval and sets it. This base object invokes `getHeaders` and sends whatever it gets back with the request. This class also provides a `getResponse` method to handle the response from the cloud provider.

If you create a custom CAPI API, you should extend `CloudRestAPIInvoker` to make REST calls to the cloud provider.

AmazonWebServicesAPIInvoker

This class is extended from `CloudRestAPIInvoker` and provides methods for making REST API calls to the cloud provider.

AzureAPIInvoker

This class is extended from `CloudRestAPIInvoker` and provides functions that make these REST API calls to Azure Resource Manager:

- Sets the Service Principal, headers, and parameters for REST calls.
- Gets the endpoint URL, token (client ID, tenant ID, and secret key), and headers/parameters.
- Retrieves and decodes the JSON response body from the cloud provider.

AzureEAAPIInvoker

This class returns the token for access to the Azure Enterprise Agreement (EA) REST API.

AzureStorageAPIInvoker
This class makes REST calls for Azure storage. The functions in this class can get the token, response, headers, and parameters.

This graphic illustrates how these base classes are used to extend AWS-specific and Azure-specific classes:

- **AmazonWebServicesAPIInvoker**
- **CloudRESTAPIInvoker**
- **AzureAPIInvoker**
- **CloudAPIBase**

### Exceptions

**RESTAPIException**
This base class handles exceptions that might occur during the REST call. It returns the response code from the cloud provider.

**AWSAPIException**
This class handles exceptions that might occur during the REST call.

**AzureAPIException**
This class throws a runtime exception when needed during Azure API invocation.
Handling exceptions

Classes for credentials

CloudRESTRequestSigningUtil
This class uses overrides that are specified in CAPI API Config Overrides. The AmazonWebServicesRequestSigner class extends this class. For Azure, many signers are built into the MID Server script includes for Method Mappers in Azure CAPI APIs.

AmazonWebServicesRequestSigner
This class sets up a secure session for the REST API call by taking information about a request, and creating a signature v4-signed string for the request. This class is extended from CloudRESTRequestSigningUtil.

Other Azure classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AzureAlertRule</td>
<td>Class to create and delete Azure alert rules.</td>
</tr>
<tr>
<td>AzureAuthorizationPolicy</td>
<td>Class to assign tag policies.</td>
</tr>
<tr>
<td>AzureComputeAvailabilitySet</td>
<td>Class to fetch Azure availability sets. Refer to: <a href="https://docs.microsoft.com/en-us/rest/api/compute/availabilitysets">https://docs.microsoft.com/en-us/rest/api/compute/availabilitysets</a></td>
</tr>
<tr>
<td>AzureComputeHardware</td>
<td>Class to fetch Azure sizes. Refer to: <a href="https://docs.microsoft.com/en-us/rest/api/compute/virtualmachinesizes">https://docs.microsoft.com/en-us/rest/api/compute/virtualmachinesizes</a></td>
</tr>
<tr>
<td>AzureComputetImage</td>
<td>Class to handle Azure API calls against image. Refer to:</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AzureDeployment</td>
<td>Class to handle Azure ARM template deployment.</td>
</tr>
<tr>
<td>AzureDeploymentOperation</td>
<td>Class to handle Azure ARM template deployment operation.</td>
</tr>
<tr>
<td>AzureImages</td>
<td>Class to handle Azure Images.</td>
</tr>
<tr>
<td>AzureLocalNetworkGateways</td>
<td>Class to create and retrieve local network gateways.</td>
</tr>
<tr>
<td>AzureNetworkRouter</td>
<td>Class to handle Azure API calls against route table. Refer to: <a href="https://docs.microsoft.com/en-us/rest/api/network/route-tables">https://docs.microsoft.com/en-us/rest/api/network/route-tables</a></td>
</tr>
<tr>
<td>AzureNetworkVirtualNetwork</td>
<td>Class to handle Azure API calls against virtual networks.</td>
</tr>
<tr>
<td>AzureProvider</td>
<td>Class to get Azure provider details, such as the provider URL and version.</td>
</tr>
<tr>
<td>AzureRegion</td>
<td>Class to fetch Azure regions.</td>
</tr>
<tr>
<td>AzureResource</td>
<td>Deals with the object referenced by ObjectID in the parameters. It also fetches the API version. This class is intended mostly to get unsupported types or delete resources.</td>
</tr>
<tr>
<td>AzureResourceGroupManager</td>
<td>Class to handle Azure ARM resource groups.</td>
</tr>
<tr>
<td>AzureResponseUtil</td>
<td>Class to translate Azure API response to DTO object.</td>
</tr>
<tr>
<td>AzureStorageBlob</td>
<td>Class to create and manipulate storage containers and blobs.</td>
</tr>
<tr>
<td>AzureStorageManagedDisk</td>
<td>Class to handle Azure API calls against managed disk. Refer to: <a href="https://docs.microsoft.com/en-us/rest/api/manageddisks/disks/disks-create-or-update">https://docs.microsoft.com/en-us/rest/api/manageddisks/disks/disks-create-or-update</a></td>
</tr>
<tr>
<td>AzureSubscription</td>
<td>Class to fetch Azure subscriptions.</td>
</tr>
<tr>
<td>AzureVirtualNetworkGateways</td>
<td>Class to create and manipulate virtual network gateways.</td>
</tr>
</tbody>
</table>
### AzureCloudEAAPIBase
Base object for Azure EA API handlers. Azure EA APIs have a base URL of https://consumption.azure.com and require the enrollment number in the path. This class handles marshalling of the EnrollmentNumber and EnrollmentToken parameters.

### Other AWS classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AmazonResponseProcessorUtil</td>
<td>Class to handle AWS REST API response.</td>
</tr>
<tr>
<td>AWSAutoScalingGroups</td>
<td>Class to handle AWS API calls to create and manipulate Auto Scaling Groups.</td>
</tr>
<tr>
<td>AWSCloudFormation</td>
<td>Handler for AWS CloudFormation API calls.</td>
</tr>
<tr>
<td>AWSEC2</td>
<td>Handler for AWS EC2 API calls.</td>
</tr>
<tr>
<td>AWSKeyValuePair</td>
<td>Class to import and manipulate key pairs.</td>
</tr>
<tr>
<td>AWSS3</td>
<td>Handler for AWS S3 API calls.</td>
</tr>
<tr>
<td>AWSTopics</td>
<td>Class with AWS topics.</td>
</tr>
</tbody>
</table>

### Azure VM provisioning walkthrough

This example walks you through the components of Cloud Management that function during the provisioning of a virtual machine in an Azure datacenter. Topics covered include blueprints, resource blocks, the Cloud API (CAPI), and MID Server script includes.

**Role required:** admin

This walkthrough starts with a Linux VM that a user already provisioned in Azure. Next, it walks you through the blueprint with the VM, the resource blocks, and then the CAPI calls specified from the resource blocks. Finally, the walkthrough shows you how a MID Server script include eventually, through several CAPI calls, makes the actual REST calls to the Azure API.

This example uses default resource blocks and script includes that are available in your instance. Therefore, although you might not have a provisioned VM on your instance, you can still follow these steps and view the components used in this example to understand how the components work.

For an example of how a custom CAPI API can be used with custom resource blocks, see the following topics:

- Create a Google Cloud Platform scoped application for Cloud Management
- Create a resource block for Google Cloud Platform
- Sample script includes for Google Cloud Platform

For an example of a VM in AWS, see AWS VM provisioning walkthrough.
Note: The terms virtual machine, VM, and virtual server are used interchangeably in this example.

1. Look at a provisioned VM in the Cloud User Portal, and focus on some of the details about the VM:
   a) In the base system instance interface, open the Cloud User Portal (Cloud Management > Cloud User Portal).
   b) On the Cloud User Portal, view a provisioned VM in a stack, such as this VM, by clicking Stacks, and then clicking the name of the stack.
      In this example, the stack is named MyAzureVM.
   c) Under Resources, click the VM in the stack.
      
      ![Virtual Server]
      
      Virtual Server
      Virtual Server
      State On

   d) View the properties of the VM, and notice that it is in an Azure Datacenter in region eastus.
2. Look at the blueprint on which the VM is based:
   a) Navigate to Design > Blueprints, and then open a blueprint with a virtual server in the Azure datacenter. The Deployment Model tab appears by default, showing you the various components of the blueprint.

   This example blueprint has three components: the container, the virtual server, which is the actual VM that is provisioned, and the Azure datacenter.
b) Click the **Operations** tab on the bottom, and then click **Provision**.

The Provision operation is the operation that the system triggered when it created the VM. Other default operations are available, but this example focuses on the Provision operation.

c) Click the **Provision** block for Blueprint Container Resource.
d) On the right, notice that one of the parameters in the Inputs list is Location. This parameter holds the value eastus, which is where the VM lives in the datacenter.

Inputs can be specified on the container, as it is in this example, or on any other resource block. By default, the Location parameter is already specified for you in the Blueprint Container resource block, so that you can use it in every blueprint like this one. If you switch the blueprint to Draft, you can add more parameters to the Blueprint Container resource block. You cannot add parameters to the Virtual Server resource block. For this walkthrough, no additional parameters are necessary.

3. View the Virtual Server and Datacenter resource blocks used in this blueprint:
   a) In the Cloud Admin Portal, navigate to Design > Resource Blocks.
   b) Search for and open the Virtual Server resource block, which is provided by default with Cloud Management.
c) On the Resource Block form, click the Host Resource related list to view the list of supported host resource blocks.

d) Notice the host interface and host resource that are already specified:
• The **Host Interface** field specifies the interface that must match the guest interface in the resource block that hosts this virtual machine. In this case, the host interface is the Compute Interface, which is also the guest interface on the Azure Datacenter resource block. By default, the datacenter resource blocks provide several guest interfaces that other resource blocks like virtual storage can use to connect to the datacenter.

• The **HostResource** column in the Host Resource related list already specifies **Azure Datacenter**, which means that the Azure datacenter resource block is a valid host for this VM.

e) Click the **Operations** tab, and then click the **Steps** subtab.

f) Select **Provision** from the **Operation** list.

Remember that the Provision operation is the operation that the system used to create the VM. Other default operations are available, but this example focuses on the Provision operation.

g) Notice the step that appears in the list and the full step description that appears above the input parameters:

• **Host Resource Operation**: indicates that this step calls an operation on the host resource (the Azure datacenter in this example).

• **Compute Interface** is the guest interface on the Azure datacenter that this step is using.

• **ConnectAndCreateVirtualMachine** is the operation in the Azure datacenter that this step calls.
Note: In this case, the step calls an operation from another resource block: the Azure datacenter. Steps can also call CAPI directly, and then CAPI can execute REST calls to the cloud provider API. You can see that when you look at the Azure datacenter resource block.

h) Navigate back to Design > Resource blocks.

i) Open the Azure Datacenter resource block, which is the host resource block that the virtual server is connected to.

j) Notice the supported guest interfaces in the Guest Interface related list:

These guest interfaces are the interfaces that the Azure datacenter makes available to other resource blocks. The Compute Interface is provided so that the Virtual Server...
resource block, which specifies the Compute Interface as its host interface, can connect to the datacenter.

k) Click the Operations tab, and then click the Steps subtab.

![Virtual Server Resource Block](image)

l) In the Interface list, select Compute Interface if it is not already selected.

Remember that this interface is specified in the Virtual Server resource block.

m) In the Operation list, search for and select ConnectAndCreateVirtualMachine.

![Operation List](image)

Remember that this operation is specified in the Virtual Server resource block.

n) Notice the CAPI call that is used in the only step for the ConnectAndCreateVirtualMachine operation:
- **Cloud API**: indicates that this step calls CAPI, so that CAPI can execute a REST call to the cloud provider, which in this case is Azure.
- **Compute Interface** specifies the CAPI interface that this step calls.
- **CreateNode** indicates the method that is executed. As the name suggests, this method tells the cloud provider to create the virtual machine.

**o)** To open the step configuration: With the resource block in the **Draft** state, point to the highlighted (blue) step, and then click the **Edit Step** icon that appears.

**p)** Look at the step configuration and notice the settings that integrate with CAPI:
Edit Operation Steps

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td>Invoke Cloud API specifies that this step should call the CAPI via the specified provider, interface, and method.</td>
</tr>
<tr>
<td>API Provider (which is actually the product)</td>
<td>Microsoft.Compute is a product (not the actual provider) that belongs to the Azure provider as defined in CAPI.</td>
</tr>
<tr>
<td>API Interface</td>
<td>Compute Interface is a product that belongs to the Azure provider as defined in CAPI.</td>
</tr>
<tr>
<td>API Method</td>
<td>CreateNode is the method that calls Azure to create the VM.</td>
</tr>
</tbody>
</table>

Note: The CAPI API definition ties together the provider (Azure), the product (Microsoft.Compute), the interface (Compute Interface), and the method (CreateNode).

a) Close the window.

r) With the Compute Interface.CreateNode step selected, click the Response Processor tab, and notice the Create_Virtual_Server_Response_Processor script.
This script is the response processor that updates the CMDB in your instance after the virtual machine is created.

s) For an explanation of the script, see Create a Response Processor and the example that is part of the topic. When you finish reviewing that topic, return to this topic.

4. Look at the CAPI components that work together to provision the VM:
   a) In the Cloud Admin Portal, navigate to Design > Cloud API.
   b) Click the API tab and then search for an open Azure Compute API.
   c) Look at how this CAPI API ties together an interface and a product:
Cloud API Name: Azure Compute API

Cloud API Interface: Compute Interface

Connector: Script Connector

Version: 1.0

Product: Microsoft.Compute

Service Category: Compute

Description: API that works with azure for compute resources

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud API Interface</td>
<td>Compute Interface is the same interface that you can see specified in the resource block. The interface contains the definition for methods, including the CreateStack method.</td>
</tr>
<tr>
<td>Connector</td>
<td>Script Connector indicates that this CAPI provides a MID Server script include for each operation. The script include calls the API. You can modify the script include or even create a new one.</td>
</tr>
<tr>
<td>Version</td>
<td>1.0 indicates the version of the API. You could have multiple versions of this API with different version numbers. Remember that in the step in the datacenter resource block that creates the VM, a Version field is provided. Although it was blank in this example because there is only one version of this API, you can specify a different version number.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Product</td>
<td><strong>Microsoft.Compute</strong> is the product that belongs to the provider in CAPI. This product is provided by default with Cloud Management, and is one of the most commonly used products for creating VMs on Azure.</td>
</tr>
</tbody>
</table>

d) In the CAPI Method Mappers related list, click the record preview icon for the **CreateNode** record, and then click **Open Record**.

e) Look at the **CreateNode** method mapper:
Notice that the Endpoint operation is Execute Script, which indicates that a customizable script include is used to call functions in CAPI, which eventually makes the REST calls to Azure. The azure-compute-1.0-CreateNode MID Server script include is specified in the Request script field.

This graphic illustrates how the components — from the resource block to the script include — are connected:
Click the record icon ( ) for the Request script. The azure-compute-1.0-CreateNode MID Server script include opens.

Review the contents of the script.

```javascript
createNode();
function createNode(){
  var acp = new AzureComputeVirtualMachine(this.parameters,
      this.headers);
  var vm = acp.createVirtualMachine();

  return vm;
}
```

The script calls the createVirtualMachine function in the AzureComputeVirtualMachine class. It returns the virtual machine that is created. The AzureComputeVirtualMachine class extends AzureCloudAPIBase, which is one of the base APIs in Cloud Management.

To see the AzureComputeVirtualMachine class, which is a MID Server script include, open the base system instance interface (not the Cloud Admin Portal), and then navigate to MID Server > Script Includes.

Search for and open AzureComputeVirtualMachine.
Review the `createVirtualMachine` function:

The function defines many variables that are required to create a VM, including the `Location` (datacenter) and the name of the VM, and so on.

```javascript
var location = this.parameters.get('Location');
var vmName = this.parameters.get('NodeName');
...
```

This line of code calls the `_performVMOperation` function in this same script include and passes in the parameters that are necessary. Notice that an empty value '' is passed for the `action` parameter:

```javascript
var response = this._performVMOperation(subscriptionId, resourceGroup,
vmName, '', 'PUT', jsonObj.encode(params));
```

The `_performVMOperation` function calls the function `_invokeRequest`, which is located in the `AzureCloudAPIBase` class:

```javascript
return this._invokeRequest(this.pathDefault, param, method,
requestBody);
```

The PUT method for the REST call is specified in the `method` parameter.

Look at the `AzureCloudAPIBase` MID Server script include. The `_invokeRequest` function first calls the `_getEndpoint` function to get the endpoint and the `_getAPIInvoker` function to identify the `AzureAPIInvoker` class as the class that invokes the actual PUT call:

```javascript
var apiInvoker = this._getAPIInvoker(apiEndpoint, this.apiVersion);
```

`AzureAPIInvoker` is extended from `CloudRESTAPIInvoker`, which provides the base functions for all invokers. Once the correct invoker class is identified, a `response` variable is created to hold the value returned from `invokePut` function in the `CloudRESTAPIInvoker`:

```javascript
response = apiInvoker.invokePut(requestBody);
```

Look at the `CloudRESTAPIInvoker`, which is the base class for all invokers. The `invokePut` function specifies the PUT method and calls `getRequest`, and it returns the response from the `getResponse` function, which gets a status code and the response from Azure. This invoker class is where the connection to Azure is made.

Return to the `AzureComputeVirtualMachine` MID Server script include and find these lines:

```javascript
var response = this._performVMOperation(subscriptionId, resourceGroup,
vmName, '', 'PUT', jsonObj.encode(params));
this._compareVMState(response.id, 'PowerState/running');
var nodeResponse = this.getVirtualMachine(response.id);
```

Remember that the variable `response` is returned from the call to `_performVMOperation`, which has the response from the API invoker that triggered the PUT method. The `nodeResponse` variable holds the response from the `getVirtualMachine` function, which is passed the ID of the virtual machine that is in the response. The `getVirtualMachine` function calls `_getVMInfo`, which calls the same `_invokeRequest` function that was called earlier. This call is another connection to the invoker, which performs the actual
REST calls. In this case, the REST call is a GET REST call to Azure to obtain the identify of the VM.

The `return nodeResponse;` line returns `nodeResponse` in the `createVirtualMachine` function, which was called from the MID Server script included on the `createNode` operation.

This graphic illustrates the classes involved:

k) Navigate back to the Azure Compute API form.

l) Click the API Config Overrides related list, and review the items that are required for authentication.

<table>
<thead>
<tr>
<th>Config Parameter and Override Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClientID $(CloudCredential.client_id)</td>
<td>The client ID is necessary to identify the application.</td>
</tr>
<tr>
<td>EndpointURL $(CloudCredential.URL)</td>
<td>The endpoint URL is the authorizing input in Azure.</td>
</tr>
<tr>
<td>SecretKey $(CloudCredential.secret_key)</td>
<td>The secret key is used to authorize the client.</td>
</tr>
<tr>
<td>TenantID $(CloudCredential.tenant_id)</td>
<td>The tenant ID is the globally unique identifier for your tenant.</td>
</tr>
<tr>
<td>Config Parameter and Override Value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>EnrollmentNumber</td>
<td>$(CloudCredential.sn_cmp_ea_credential.enrollment_number)</td>
</tr>
<tr>
<td>EnrollmentToken</td>
<td>$(CloudCredential.sn_cmp_ea_credential.access_key)</td>
</tr>
</tbody>
</table>

See *Store the Azure service principal credentials in the instance* for more information on the Azure information that is used in these overrides.

m) Navigate back to Design > Cloud API, and then click the Interface tab.

n) Search for and open Compute Interface.

o) Review the contents of the interface.

Notice that the interface provides REST response structures for methods like CreateNode. You typically do not need to modify existing interfaces.
Note the service category and the operations:

<table>
<thead>
<tr>
<th>Field or related list</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Category</td>
<td>The service category classifies the interface. The category for the Compute Interface is also <strong>Compute</strong>.</td>
</tr>
<tr>
<td>CAPI Interface Operations</td>
<td>The interface operations define the JSON structure for the REST call and the parameters that are required for the interface.</td>
</tr>
</tbody>
</table>

p) Click the **CreateNode** CAPI interface operation. This operation is the operation that provides the framework for creating the EC2 virtual server.

q) Review the components of the operation:
<table>
<thead>
<tr>
<th>Field or related list</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response structure</td>
<td>The response structure is the JSON framework for REST API call. It provides a list of attributes that the provider can use to create the virtual resource with empty values.</td>
</tr>
<tr>
<td>Interface Operation Parameters</td>
<td>These parameters are also the parameters that the CAPI interface needs from the system so it can pass it in the REST call to the cloud provider.</td>
</tr>
</tbody>
</table>
This record is in the **Cloud API application**, but **Global** is the current application. To edit this record, you need to change the application.

**Operation name**: CreateNode

**Cloud API Interface**: Compute Interface

**Response structure**:

```json
{
    "nodeId": "",
    "nodeName": "",
    "imageld": "",
    "nodeGroup": "",
    "state": "",
    "location": "",
    "scope": "",
    "availabilityZone": "",
    "hostName": "",
    "hostServer": "",
    "osName": "",
    "osVersion": "",
    "osVendor": "",
    "userId": "",
    "userName": "",
    "privateKey": "",
    "password": "",
    "hardwareId": "",
    "hardwareProvider": "",
    "ram": "",
    "processors": [{
        "cores": "",
        "speed": ""
    }]
}
```
r) Navigate back to **Design > Cloud API** and then click the **Provider** tab.

s) Open the **Azure** provider.

   The Azure Cloud Provider form opens, showing you that this provider is based on an existing CMDB class: **Azure Datacenter (cmdb_ci_azure_datacenter)**.

t) Click the Cloud Products related list, and sort the list by the **Name** column.

Notice that several Azure products are available by default. One of the most commonly used Azure products created from templates is **Microsoft.Compute**.

u) Click **Microsoft.Compute** in the **Name** column. Notice that the product specifies many resource types, each of which is mapped to a CI class.
These resource types indicate some of the CIs, but not all, that are related to the virtual machine. The response processor in the resource block populates CIs with data when the cloud provider provisions the VM.

5. To see the important CIs that are related to the VM:
a) On the Cloud User Portal, click **Stacks**, and then open the stack containing the VM.

b) Click the **View Dependency** icon.

The dependency map displays the stack CI at the top, the VM in the middle, and the various related CIs, such as the image, at the bottom.

c) To view the form for the VM in the CMDB, right-click the arrow next to any CI.
d) From the menu, select **View Form**.

The CI form opens, showing you much of the information is already available on the Cloud User Portal when you view the properties of the VM.
The VM CI form

VM properties in the Cloud User Portal

<table>
<thead>
<tr>
<th>Name</th>
<th>MyVMNodName</th>
</tr>
</thead>
<tbody>
<tr>
<td>State</td>
<td>On</td>
</tr>
</tbody>
</table>

Object ID: /subscriptions/c.../providers/Micro.../subscriptions/c...

Attached Volume ID: /subscriptions/c...

Cloud Account: Hybrid Cloud East

Created On: 2018-09-17 17:36:42

Disks: 1

Instance ID: /subscriptions/c...

Name: MyVMNodName

OS: /Canonical/Ubuntu...

Password: Show Password

Private IP: 10.0.0.101

Provider: Azure Datacenter

Public IP: 137.135.70.200

Region: eastus

ResourceGroup Name: rg-800817

Size: Basic_A0

State: On

Type: Virtual Machine Instance

UserName: cloudd

Tags:

Stack: MyAzureVMStack

User: System Administrator

User Group: Team Development Code Review
AWS VM provisioning walkthrough

This example walks you through the components of Cloud Management that function during the provisioning of a virtual machine in an AWS datacenter. Topics covered include blueprints, resource blocks, the Cloud API (CAPI), and MID Server script includes.

Role required: admin

This walkthrough starts with a Windows VM that a user already provisioned in AWS. Next it walks you through the blueprint with the VM, the resource blocks, and then the CAPI calls specified from the resource blocks. Finally, the walkthrough shows you how a base MID Server script include makes the actual calls to the AWS API.

This example uses default resource blocks and script includes that are available in your instance. Therefore, although you might not have a provisioned VM on your instance, you can still follow these steps and view the components used in this example to understand how the components work.

For an example of a VM in Azure, see Azure VM provisioning walkthrough.

Note: The terms virtual machine, VM, and virtual server are used interchangeably in this example.

1. Look at a provisioned VM in the Cloud User Portal, and focus on some of the details about the VM:
   a) On your instance, open the Cloud User Portal (Cloud Management > Cloud User Portal).
   b) View a provisioned VM in a stack, such as this Windows VM, by clicking Stacks, and then clicking the name of the stack.
      In this example, the stack is named MyStack.

   c) Under Resources, click the VM in the stack.
d) View the properties of the VM, and notice that it is in region us-east-1 in an AWS Datacenter.
2. Look at the blueprint on which the VM is based:
   a) Navigate to Design > Blueprints, and then open a blueprint with a virtual server in the Azure datacenter. The Deployment Model tab appears by default, showing you the various components of the blueprint.

   This example blueprint has three components: the container, the virtual server, which is the actual VM that is provisioned, and the AWS datacenter.
b) Click the **Operations** tab on the bottom, and then click **Provision**.

The Provision operation is the operation that the system triggered when it created the VM. Other default operations are available, but this example focuses on the Provision operation.

c) Click the **Provision** block for Blueprint Container Resource.
d) On the right, notice that one of the parameters in the Inputs list is Location. This parameter holds the value eastus, which is where the VM lives in the datacenter.

Inputs can be specified on the container, as it is in this example, or on any other resource block. By default, the Location parameter is already specified for you in the Blueprint Container resource block, so that you can use it in every blueprint like this one. If you switch the blueprint to Draft, you can add more parameters to the Blueprint Container resource block. You cannot add parameters to the Virtual Server resource block. For this walkthrough, no additional parameters are necessary.

3. Look at the Virtual Server and AWS Datacenter resource blocks used in this blueprint:
   a) In the Cloud Admin Portal, navigate to Design > Resource Blocks.
   b) Search for and open the Virtual Server resource block, which is provided by default with Cloud Management.
c) On the Resource Block form, click the **Host Resource** related list to view the list of supported host resource blocks.

d) Notice that the host interface and host resource are already specified:
• The **Host Interface** field specifies the interface that must match the guest interface in the resource block that hosts this virtual machine. In this case, the host interface is the Compute Interface, which is also the guest interface on the AWS Datacenter resource block. By default, the datacenter resource blocks provide several guest interfaces that other resource blocks like virtual storage can use to connect to the datacenter.

• The **HostResource** column in the **Host Resource** related list already specifies **AWS Datacenter**, which means that the AWS datacenter resource block is a valid host for this VM.

e) Click the **Operations** tab, and then click the **Steps** subtab.
f) Select **Provision** from the **Operation** list.

Remember that the Provision operation is the operation that the system used to create the VM. Other default operations are available, but this example focuses on the Provision operation.

g) Notice the step that appears in the list and the full step description that appears above the input parameters:

- **Host Resource Operation**: indicates that this step calls an operation on the host resource (the AWS datacenter in this example).
- **Compute Interface** is the guest interface on the AWS datacenter that this step is using.
- **ConnectAndCreateVirtualMachine** is the operation in the AWS datacenter that this step calls.

**Note**: In this case, the step calls an operation from another resource block: the AWS datacenter. Steps can also call CAPI directly, and then CAPI can execute REST calls to the cloud provider API. You can see that when you look at the AWS datacenter resource block.

h) Navigate back to **Design > Resource blocks**.

i) Open the **AWS Datacenter** resource block, which is the host resource block that the virtual server is connected to.

j) Notice the supported guest interfaces in the **Guest Interface** related list:
These guest interfaces are the interfaces that the AWS datacenter makes available to other resource blocks. The **Compute Interface** is provided so that the Virtual Server resource block, which specifies the Compute Interface as its host interface, can connect to the datacenter.

k) Click the **Operations** tab, and then click the **Steps** subtab.
l) In the **Interface** list, select **Compute Interface** if it is not already selected.

Remember that this interface is specified in the Virtual Server resource block.

m) In the **Operation** list, search for and select **ConnectAndCreateVirtualMachine**.

Remember that this operation is specified in the Virtual Server resource block.

n) Notice the CAPI call that is used in the only step for the **ConnectAndCreateVirtualMachine** operation:
- **Cloud API**: indicates that this step calls CAPI, so that CAPI can execute a REST call to the cloud provider, which in this case is AWS.

- **Compute Interface** specifies the CAPI interface that this step calls.

- **CreateNode** indicates the method that is executed. As the name suggests, this method tells AWS to create the virtual machine.

o) With the resource block in the Draft state, point to the highlighted (blue) step and then click the **Edit Step** icon to open the step configuration.

p) Look at the step configuration, and notice the settings that integrate with CAPI:
Edit Operation Steps

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation Type</td>
<td><strong>Invoke Cloud API</strong> specifies that this step should call the CAPI via the</td>
</tr>
<tr>
<td></td>
<td>given provider, interface, and method.</td>
</tr>
<tr>
<td>API Provider</td>
<td><strong>AWS Elastic Compute Cloud</strong> is a product (not that actual provider) that</td>
</tr>
<tr>
<td></td>
<td>belongs to the AWS provider as defined in CAPI.</td>
</tr>
<tr>
<td>API Interface</td>
<td><strong>Compute Interface</strong> is a product that belongs to the AWS provider as</td>
</tr>
<tr>
<td></td>
<td>defined in CAPI.</td>
</tr>
<tr>
<td>API Method</td>
<td><strong>CreateNode</strong> is the method that calls AWS to create the VM.</td>
</tr>
</tbody>
</table>

**Note:** The CAPI API definition ties together the provider (AWS), the product (AWS Elastic Compute Cloud), the interface (Compute Interface), and the method (CreateNode).
q) Close the window.

r) With the **Compute Interface.CreateNode** step selected, click the **Response Processor** tab, and notice the **Create_Virtual_Server_Response_Processor** script.

This script is the response processors that updates the CMDB in your instance after the virtual machine is created in AWS.

s) For an explanation of this script, see *Create a Response Processor* and the example that is part of that topic. After you finish reviewing that topic, come back to this one.

4. Look at the CAPI components that work together to provision the VM:
   a) In the Cloud Admin Portal, navigate to **Design > Cloud API**.
   b) Click the **API** tab.
   c) Search for an open **AWS Compute API**.
      There are two AWS Compute API records. Open the first one in the list that matches the image in the next step.
   d) Look at how this CAPI API ties together an **interface** and a **product**:
Field | Description
--- | ---
Cloud API Interface | **Compute Interface** is the same interface that you can see specified in the resource block. The interface contains the definition for methods, including the **CreateNode** method.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connector</td>
<td><strong>Cloud Compute Connector</strong> indicates that this CAPI makes Java calls to the system, which then calls the AWS API. This connector is not scripted. If this were a scripted connector, it was have a MID Server script include that calls the AWS API.</td>
</tr>
<tr>
<td>Version</td>
<td>1.0 indicates the version of the API. You could have multiple versions of this API with different version numbers. Remember that in the step in the AWS Datacenter resource block that creates the VM, a Version field is provided. Although it was blank in this example because there is only one version of this API, you could specify a different version number.</td>
</tr>
<tr>
<td>Product</td>
<td><strong>AWS Elastic Compute Cloud</strong> is the product that belongs to the AWS provider in CAPI. This product is provided by default with Cloud Management, and is one of the most commonly used products for creating VMs on AWS.</td>
</tr>
</tbody>
</table>

**Note:** Most AWS-related APIs that are provided in the Cloud Management application by default are not scripted, like this one, and cannot be modified. But you can create your own scripted APIs.

e) In the CAPI Method Mappers related list, click the record preview icon for the **CreateNode** record, and then click **Open Record**.
f) Look at the **CreateNode** method mapper:
Notice that the Endpoint operation is not **Execute Script**. This value indicates that the CreateNode method is using a Java call within the Cloud Management application on your instance to make a REST call to the AWS API. Therefore, you cannot modify how the CreateNode method works. If the value was **Execute Script**, you would see a MID Server script include specified in the **Request** script field. You would be able to modify that script include, or specify a new one, to customize the REST calls to the AWS provider.

g) Scroll down and look at parameters in the CAPI Parameter Mappers related list. Notice that important parameters, such as **Location**, are provided.

h) Navigate back to the AWS Compute API form.

i) Click the **API Config Overrides** related list, and review the items that are required for authentication:

<table>
<thead>
<tr>
<th>Config Parameter and Override Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccountAliasName $(CloudCredential.Alias)</td>
<td>The account alias is an optional value that you can create in your AWS account. It is a secondary name for your account ID. See the AWS documentation for more information.</td>
</tr>
<tr>
<td>Config Parameter and Override Value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Credentials $(CloudCredential.secret_key)</td>
<td>The secret key is used with the access key for authentication. You configured this value in your AWS credential record during setup. To refer to that procedure, see <a href="#">Create the credentials that enable Cloud Management to access your AWS data</a>.</td>
</tr>
<tr>
<td>Endpoint $(CloudCredential.URL)</td>
<td>The endpoint is the URL that your users must log in to and authenticate. It uses the account ID or the account alias. See the <a href="#">AWS documentation</a> for more information.</td>
</tr>
<tr>
<td>Identity $(CloudCredential.access_key)</td>
<td>The Identify record holds the AWS access key, which AWS requires for authentication.</td>
</tr>
</tbody>
</table>

j) Navigate back to Design > Cloud API, and then click the Interface tab.

k) Search for and open Compute Interface. This interface is the interface specified in the resource block and the AWS Compute API CAPI API.

l) Review the contents of the interface.
Notice that the interface provides REST response structures for methods like `CreateNode`. You typically do not need to modify existing interfaces.

Note the service category and the operations:

<table>
<thead>
<tr>
<th>Field or related list</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Category</td>
<td>The service category classifies the interface. The category for the Compute Interface is also <code>Compute</code>.</td>
</tr>
<tr>
<td>CAPI Interface Operations</td>
<td>The interface operations define the JSON structure for the REST call and the parameters that are required for the interface.</td>
</tr>
</tbody>
</table>

m) Click the `CreateNode` CAPI interface operation. This operation is the one that provides the framework for creating the EC2 virtual server in AWS.
n) Review the components of the operation:

<table>
<thead>
<tr>
<th>Field or related list</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response structure</td>
<td>The response structure is the JSON framework for REST API call. It provides a list of attributes that AWS can use to create the virtual resource with empty values.</td>
</tr>
<tr>
<td>Interface Operation Parameters</td>
<td>These parameters are also the parameters that the CAPI interface needs from the system so it can pass it in the REST call to the cloud provider.</td>
</tr>
</tbody>
</table>
This record is in the **Cloud API application**, but **Global** is the current application. To edit this record, **Global** must be the current application.

**Operation name**: CreateNode

**Cloud API Interface**: Compute Interface

**Response structure**:

```json
{
    "nodeId": "",
    "nodeName": "",
    "imageId": "",
    "nodeGroup": "",
    "state": "",
    "location": "",
    "scope": "",
    "availabilityZone": "",
    "hostName": "",
    "hostServer": "",
    "osName": "",
    "osVersion": "",
    "osVendor": "",
    "userId": "",
    "userName": "",
    "privateKey": "",
    "password": "",
    "hardwareId": "",
    "hardwareProvider": "",
    "ram": "",
    "processors": [{
        "cores": "",
        "speed": ""
    }
}
```
o) Navigate back to **Design > Cloud API**, and then click the **Provider** tab.

p) Open the **AWS** provider.

The AWS Provider form opens, showing you that this provider is based on an existing CMDB class: **AWS Datacenter (cmdb_ci_aws_datacenter)**.

q) Click the Cloud Products related list if it is not already selected, and sort the list by the **Name** column.

Notice that several AWS products are already available by default. One of the most commonly used AWS products is **Elastic Compute Cloud (EC2)**.
r) Click **AWS Elastic Compute Cloud** in the **Name** column. Notice that the product specifies many resource types, each of which is mapped to a CI class.
These resource types indicate some of the CIs, but not all, that are related to the virtual machine. The response processor in the resource block populates CIs with data when AWS provisions the VM.

5. To see the important CIs that are related to the VM:
   a) On the Cloud User Portal, click **Stacks**, and then open the stack containing the VM.
   b) Click the View Dependency icon.
The dependency map displays the stack CI at the top, the VM in the middle, and the various related CIs, such as the image, at the bottom.

c) To view the form for the VM in the CMDB, right-click the arrow next to any CI, such as the VM.

d) From the menu, select View Form.
The CI form opens, showing you that much of the information is already available on the Cloud User Portal when you view the properties of the VM.
Add a product to an existing provider in CAPI

The Cloud Management Cloud API (CAPI) offers many providers, such as AWS and Azure, and products, such as EC2 instances. However, you can add products to existing providers for new types of cloud resources.

- Role required: admin
- Review Default CAPI APIs and make sure the provider and product that you want to create are not already provided.
Familiarize yourself with the cloud provider API so that you know how to create REST calls to the provider. If you are not familiar with the headers, authentication requirements, endpoint, and data that you can pass into the provider API, you cannot build your own CAPI components.

This topic provides general instructions for extending an existing cloud provider, product, and resource type. For a complete example, see Create CI class provider for Google Cloud Platform and Create a CI class for the Puppet provider. These examples are part of the Scoped applications for Cloud Management scenarios that show you in detail how to create scoped apps. The examples also cover the configuration of CAPI, resource blocks, blueprints, and credential mappings.

1. In your instance (not in the Cloud Admin Portal), enter the following text into the application filter:

   sn_capi_product.list

2. Press ENTER.

   The list of products that belong to all providers appears. You can filter the list to verify that the product you want to create is not already provided.

3. Click New and then enter a name and choose one of the existing service categories that is most appropriate.

4. Click Submit.

   The list of products appears again.

5. Find the product that you created, and then double-click the empty cell in the Provider column.

6. Add the correct provider.

7. Add a unique provider name: Double-click the cell in the Provider Name field and then change the name.

   The system can auto-generate a provider name for you, but you should modify it to avoid conflicts with existing provider names.
8. In the Cloud Admin Portal, navigate to **Design > Cloud API**.
9. Click the tile for the provider to which you added the new product.
10. Verify that your product is in the **Cloud Products** related list.
11. Click the information icon ( ) for the cloud product, and then click **Open Record**.
12. In the Cloud Resource Types related list, click **New** and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a descriptive name.</td>
</tr>
<tr>
<td>Product</td>
<td>Verify that the resource type is for the correct product.</td>
</tr>
<tr>
<td>CI Class</td>
<td>Select a CMDB table. If you do not see the correct CMDB table, see <a href="#">Create a CI class for a virtual cloud resource</a> for instructions on how to create one.</td>
</tr>
</tbody>
</table>

You can ignore the **Pattern** and **Dependency sequence** fields.

13. Click **Submit**.

*Create a CI class for a virtual cloud resource* if necessary, or *Create or extend a CAPI interface*

Create a CI class for a virtual cloud resource

If the base system does not provide a CI type for cloud-based virtual resources, you can create a CI class based on the virtual machine object class.

Role required: admin
This topic provides general instructions for extending an existing cloud provider, product, and resource type. For a complete example, see Create CI class provider for Google Cloud Platform and Create a CI class for the Puppet provider. These examples are part of the Scoped applications for Cloud Management scenarios that show you in detail how to create scoped apps. The examples also cover the configuration of CAPI, resource blocks, blueprints, and credential mappings.

1. Navigate to Configuration > CI Class Manager.
2. Click Open Hierarchy.

3. On the CI Classes pane, enter Virtual Machine Object in the search field, and then select Virtual Machine Object.

4. Click the Virtual Machine Object link.
5. Click Add Child Class.

6. In the Provide Basic Info step that appears, enter a name and a description.

When you move the focus of the cursor away from the Display Name field, the Table Name field automatically populates with the same name in lower case. For a table in a scoped application, the name is prefixed with a namespace identifier to indicate that it is part of an application. For a table in the global application, the name is prefixed with the string u_cmdb_ci. You cannot modify the prefix; however, you can modify the rest of the table name. The name can contain only lowercase alphanumeric ASCII characters and underscores (_). The maximum string length is 80 characters.

7. Click Next.

8. On the Add Attributes step, click Derived, and then review the attributes that are inherited from the Virtual Machine Object class.

Most attributes that you need are provided.

9. Optional: To add an attribute:
   a) Double-click the Insert a new row text in the Column label field.
   b) Enter a name for the column and then click the green arrow.
   c) Fill out the rest of the fields in the row. See Dictionary Entry Form for a description of all the fields.

10. Click Next.

11. On the Set Identification Rule step, create an identification rule that specifies a unique Object ID, which is required for identification.

    Important: Cloud Management depends on the Object ID to uniquely identify an instance of the CI Class Type.

    a) Click Replace on the default rule under Identification Rule.
    b) Enter a name for the rule and select the Dependent option.
    c) Click Save.
    d) Under Identifier Entries, click Add.
e) Move **Object ID** from the **Available** column to the **Selected** column.
f) Click **Save**.

12. Click **Next**.
13. On the **Add Dependent Relationships** step, click **Override Dependencies**.
14. Configure the relationship between this virtual resource and the logical datacenter on which it resides:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule Type</td>
<td>Hosting</td>
</tr>
<tr>
<td>Relationship</td>
<td>Hosted on (Hosted on::Hosts)</td>
</tr>
<tr>
<td>Target Class</td>
<td>Logical Datacenter</td>
</tr>
</tbody>
</table>

15. Click **Save**.
16. Click **Next**.
17. Click **Next** again to skip the **Add Reconciliation Rules** step, and then click **Done** to skip the **Add Suggested Relationships** step.

Use the CI class in a resource type for a cloud provider. See [Add a product to an existing provider in CAPI](#) for instructions.

Your response processor, which is part of the resource block, must also specify the necessary CI class types so it can update the CMDB. Usually, the response processor needs the object ID, the specific datacenter, and the cloud service account. See [Virtual server response processor example](#) for example response processor code that uses CI classes.

**Create or extend a CAPI interface**

You can create a new CAPI interface or extend an existing one by adding new operations.

- **Role required:** admin
- Review [Default CAPI APIs](#) and make sure the provider and product that you want to create do not exist.
- Familiarize yourself with the cloud provider API so that you know how to create REST calls to the provider. If you are not familiar with the headers, authentication requirements, endpoint, and data that you can pass into the provider API, you cannot build your own CAPI components.

Usually, you can use the default interfaces that are provided in the Cloud Management application. To create a custom interface, base it on an existing one that is similar.

1. In the Cloud Admin Portal, navigate to **Design > Cloud API**, and then click the **Interface** tab.
2. Click **New**.
3. In the Add Cloud interface window, enter a name for the interface, a helpful description, and select the most relevant category to which the API interface should belong.

This example shows an interface for Azure storage:

**Add Cloud Interface**

<table>
<thead>
<tr>
<th>Cloud Interface</th>
<th>Azure Block Blob Storage Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>An interface for storage storage of objects for Azure</td>
</tr>
<tr>
<td>category</td>
<td>Storage</td>
</tr>
</tbody>
</table>

4. In the CAPI Interface Operations related list, click **New**.

Each operation that you create corresponds to a different REST API call.
5. Provide a name for the operation that corresponds to the API you want to call in the provider. For example, you can create an operation called **CreateContainer** that corresponds to the Create Container operation in Azure.

6. Right-click the header and select **Save**.

7. In the Interface Operation Parameters related list, click **New**.

8. Create a parameter that the operation needs as defined by the cloud provider REST API documentation. If you do not know the parameters, the operation fails.

   In the Azure storage example, the **CreateContainer** operation must have the name of the container that is referred to here as **ContainerName**.

9. Click **Submit**.

10. Add as many parameters as you need for the operation, and then add all the operations you need.

    **Create a CAPI API**

**Create a CAPI API**

If the existing CAPI APIs do not integrate with the CAPI interface for a provider, you can create an API. The steps in resource blocks can call your API methods when operations are executed.
• Role required: admin
• Review Default CAPI APIs and make sure the provider and product that you are creating do not exist.
• Familiarize yourself with the cloud provider API so that you know how to create REST calls to the provider. If you are not familiar with the headers, authentication requirements, endpoint, and data that you can pass into the provider API, you cannot build your own CAPI components.

This topic provides instructions for creating a CAPI API. For a complete example, see Create a cloud API for Puppet and Create a cloud API for Google Cloud Platform. Both of those examples are part of the Scoped applications for Cloud Management scenarios that show you in detail how to create scoped apps. They also cover the configuration of CAPI, resource blocks, blueprints, and credential mappings.

1. In the Cloud Admin Portal, navigate to Design > Cloud API.
2. Click the API tab.
3. Click New.
4. Fill out the form fields (see table):

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud API</td>
<td>Provide a descriptive name.</td>
</tr>
<tr>
<td>Interface</td>
<td>Select from among the existing interfaces that provide the operations you need.</td>
</tr>
<tr>
<td>Connector</td>
<td><strong>Script Connector</strong> appears in this field if the connector is a Java connector that is available out-of-box and cannot be modified. You do not need to modify these script connectors.</td>
</tr>
<tr>
<td>Product</td>
<td>Select the product that is part of the provider. The product includes resource types in the CMDB.</td>
</tr>
<tr>
<td>Version</td>
<td>Provide a version number to identify this API. Enter 1.0 if this API is the first version. You can create several versions of the same API and then refer to the specific versions in the operation steps of resource blocks.</td>
</tr>
<tr>
<td>Scripted</td>
<td>This field is selected if the Interface value is Script Connector.</td>
</tr>
<tr>
<td>Script Type</td>
<td>Select the language for the script.</td>
</tr>
<tr>
<td>ExecuteScriptOnInstance</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Provide a description for this API.</td>
</tr>
</tbody>
</table>

5. Verify that all the operations you need appear in the CAPI Method Mappers related list.

The operations in the CAPI Method Mappers list are the same operations in the interface that you selected. They are bound to this API. You cannot add or remove any operations from the list. Some of the operations, especially Azure operations, use MID Server script includes

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
that perform the actual implementation for the API. The script includes a call to the REST API for the provider, such as AWS or Azure. You should be able to find the API in the provider's documentation.

6. To see the script include for a CAPI Method Mapper:
   a) Click the gear icon (⚙️) on the CAPI Method Mappers related list, add the Request Script field to the Selected column, and then click OK.

   The MID Server script includes now appear in the list of method mappers.

   b) Click the name of a script to view the script include.
   
   In this example, the `CreateNode` operation uses the `azure-compute-1.0-CreateNode` script include.
That script include contains the following code:

```javascript
createNode();
function createNode()
{
    var acp = new AzureComputeVirtualMachine(this.parameters,
        this.headers);
    //this passes the parameters and headers into the
    AzureComputeVirtualMachine script include.
    var vm = acp.createVirtualMachine();

    return vm;
}
```

Notice that the code calls the `AzureComputeVirtualMachine` script include, which is available by default with the Cloud Management application. It contains all the API calls to Azure against virtual machines.

7. Create an API config override to pass the necessary credential information to the cloud provider. You also need to pass header information required by the cloud provider REST API.
API config overrides are necessary because credential data is not included when your instance makes the REST call to the cloud provider API.

Tip: If you do not know which API config overrides to create, look at a similar API that exists in CAPI, and use the same overrides. For example, if you are creating an API for something like AWS storage, look at other AWS APIs such as AWS Block Storage or AWS Compute.

a) In the API Config Override related list, click New.

b) Click the lookup icon and select an existing config parameter. Many useful parameters are provided by default, such as Identity, Credentials, and so on.

c) Enter an Override value.

You can set a static value, or you can pull a value from the system in the form of $(CloudCredential.attribute_name). The CloudCredential portion unlocks it from the credential store. The attribute_name portion is an attribute that depends on the particular credential or alias or endpoint URL that you are trying to retrieve.

If you navigate to Connections and Credentials > Credentials and open your AWS credential, you can right-click the Access Key ID field and see the name of the column in the Credentials table where the access key is stored. In this case, the access key is stored in the access_key column. So your Override value should be $(CloudCredential.access_key).
8. If you need to create a configuration parameter to use in the API Config Override, follow this procedure:
   a) On the API Config Override form, click the lookup icon in the **Config Parameter** field.
   b) Click **New** on the Configuration Parameter window and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Config parameter name</td>
<td>Enter a descriptive name.</td>
</tr>
<tr>
<td>Default value</td>
<td>Set an optional default value.</td>
</tr>
<tr>
<td>Endpoint</td>
<td>This field is read only.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a helpful description.</td>
</tr>
<tr>
<td>Config data type</td>
<td>Select the type of data:</td>
</tr>
<tr>
<td></td>
<td>· String</td>
</tr>
<tr>
<td></td>
<td>· DateTime</td>
</tr>
<tr>
<td></td>
<td>· Date</td>
</tr>
<tr>
<td></td>
<td>· Long</td>
</tr>
<tr>
<td></td>
<td>· Password</td>
</tr>
<tr>
<td></td>
<td>· Boolean</td>
</tr>
<tr>
<td>Config type</td>
<td>Select the type of configuration:</td>
</tr>
<tr>
<td></td>
<td>· Header: Use this value to pass the override values as headers. If your have a MID Server script include as part of your API and it uses <code>this.headers</code> as a parameter, the override values are passed to the provider API.</td>
</tr>
<tr>
<td></td>
<td>· Config</td>
</tr>
<tr>
<td></td>
<td>· URI Config</td>
</tr>
</tbody>
</table>

9. If necessary, modify the script includes that execute the operations against the cloud provider API:
   a) On the Cloud API form, in the CAPI Method Mappers related list, click the information icon ( ) next to the CAPI interface operation. Do not click the CAPI Interface Operation name itself.
   b) Click **Open Record**.
   c) On the operation form, you can modify the following field values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint operation</td>
<td>Keep the default <code>Execute Script</code> endpoint operation.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Supported in version</td>
<td>You typically do not need to change this value. Change the <strong>Supported in</strong> version to use the method mapper in a different version of the API. By default, the method mappers are set to the same version number as the API they belong to.</td>
</tr>
</tbody>
</table>

**d)**

In the **Request Script** field, click the information icon ( información icon ( ) next to the placeholder MID Server script include to modify it.

**e)** On the script include form, create your script.

The following example JavaScript lists Azure load balancers. Note the helpful comments:

```javascript
getResourceInfo();

function getResourceInfo() {
  var anpi = new AzureLoadBalancer(this.parameters, this.headers);

  //this calls the 'AzureLoadBalancer' MID Server script include, which in turn
  //calls the 'AzureCloudAPIBase' script include, both of which
  //are already in your instance by default.
  //The 'AzureCloudBase' script include runs a query for
  //resources of a given type in a
  //specified region.

  var items = anpi.listResourcesByRegionOrIDs();
  //This function is in the 'AzureCloudAPIBase' script include. It calls the Azure API.

  return items;
}
```

The overrides provided for this operation are what Azure needs to find the load balancers: the Client ID, Tenant ID, the Secret Key, and the Endpoint URL:
For other examples of MID Server script includes that you can use in CAPI APIs, see Sample scripts for Cloud API interface operations and Sample script includes for Google Cloud Platform.

**Note:** For the Google Cloud integration example, a MID Server JAR file is also needed.

f) Click **Submit**.

g) Modify the script include for each operation.

### Cloud User Portal

The Cloud User Portal gives you immediate access to all day-to-day cloud activities.

**Cloud User Portal**

Use the Cloud User Portal to perform the following actions:

- Monitor your quota, costs, budget, life cycle events, stack health, and requests.
- Request stacks from the service catalog and track requests.
- Request life cycle operations for stacks and resources (for example, stop, start, or deprovision).
- Create and track incidents.
Using the `Overview` page

![Cloud User Portal](image)

<table>
<thead>
<tr>
<th>Stack Health</th>
<th>Recent Stacks</th>
<th>Quota</th>
<th>Current Month Spend</th>
</tr>
</thead>
<tbody>
<tr>
<td>MyStack-007</td>
<td>Sundar Stack</td>
<td>$3077.86</td>
<td>$284.00 Used</td>
</tr>
<tr>
<td>HP AWS Single VM Stack</td>
<td>act</td>
<td>12/28/2020</td>
<td>0/1000 VCPUs Count</td>
</tr>
<tr>
<td>HP AWS Single VM Stack</td>
<td>arjun Stack</td>
<td>12/28/2020</td>
<td>77 VM Count</td>
</tr>
<tr>
<td>TestVMmember</td>
<td>Fremont</td>
<td>12/28/2020</td>
<td>21 Network Count</td>
</tr>
<tr>
<td>sandare</td>
<td>Sundar Stack</td>
<td>12/28/2020</td>
<td></td>
</tr>
</tbody>
</table>
Tasks on the ‘Overview’ page

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Launch a Stack</td>
<td>Open the service catalog of items that you can request. You request an item and, when your request is approved, the system launches the stack. See <a href="#">Launch a stack</a>.</td>
</tr>
<tr>
<td>Manage Stacks</td>
<td>View the stacks assigned to you. See <a href="#">Manage a stack</a>.</td>
</tr>
<tr>
<td>View Activities</td>
<td>View activities for your stacks and resources. Activities include stack requests, change requests, incidents, cloud events, and SSH keys. See <a href="#">View Activities</a>.</td>
</tr>
<tr>
<td>View Dashboard</td>
<td>Opens the <a href="#">Cloud Management user dashboards</a>.</td>
</tr>
<tr>
<td>Pending Requests</td>
<td>When you submit a request for a stack, the Activities page displays the progress and status of the stack request, and a Pending Requests link and count appears. Click the link to view a summary of pending stack requests. See <a href="#">Track a stack request</a>.</td>
</tr>
</tbody>
</table>

Quick-look status reports on the ‘Overview’ page

<table>
<thead>
<tr>
<th>View</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Views</td>
<td>A filter to view and manage your resources as well as the resources assigned to the group you belong to.</td>
</tr>
<tr>
<td>Current Month Spend</td>
<td>A chart that shows costs, by category, for your cloud resources.</td>
</tr>
<tr>
<td>Budget</td>
<td>A chart displaying an arc based on your consumption of budget allocated to you.</td>
</tr>
<tr>
<td>Quota</td>
<td>For each quota-defined resource (VM count, vCPU count, Storage size, and Network count), a graph indicates the number that you have consumed compared to the maximum limit allocated to you (your quota).</td>
</tr>
<tr>
<td>Recent Stacks</td>
<td>Ten stacks that you most recently requested. Each entry shows the Name, Catalog item requested, Cloud account used to request the stack, and the order status of the requested stack. Click a Name to view stack details.</td>
</tr>
<tr>
<td>Stack Health</td>
<td>Shows the most recent 10 items: Incidents created for provisioned stacks or active stacks with a resource turned off. Also shows the resources which are down. Click an item to view stack details. Color codes for incident type:</td>
</tr>
<tr>
<td></td>
<td>• Red: Outage.</td>
</tr>
<tr>
<td></td>
<td>• Orange: Performance issue or Something is broken.</td>
</tr>
<tr>
<td></td>
<td>• Grey: Question or Request.</td>
</tr>
</tbody>
</table>

Using the task bar

From the task bar, you can perform the following actions:

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>Return to the home page of the Cloud User Portal.</td>
</tr>
<tr>
<td>Catalog</td>
<td>View the Service Catalog of items that you can request to launch stacks. When you select a Service Category (VM, Compute, Network, Storage, and so on), the All Items section lists the items in the selected category. Click Launch to request an item. See <a href="#">Launch a stack</a>.</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Stacks</td>
<td>Lists all stacks that are assigned to you. Select a catalog item to view the stacks that the item launches. Click a stack to view details—all resources, resource properties, activities, and life cycle operations. See <a href="#">Manage a stack</a>.</td>
</tr>
<tr>
<td>Resources</td>
<td>Lists individual resources of the provisioned stacks that are assigned to you, regardless of who requested the stack. See <a href="#">Manage a resource</a>.</td>
</tr>
<tr>
<td>Activities</td>
<td>Stack requests, change requests, cloud life cycle events (if configured), and incidents that are associated with your stacks and resources. SSH keys that you created or that were generated for you. See <a href="#">View Activities</a>.</td>
</tr>
<tr>
<td>Dashboard</td>
<td>Provides insight on cloud costs and usage. See <a href="#">Cloud Management user dashboards</a>.</td>
</tr>
</tbody>
</table>

**Cloud Management user dashboards**

The **Dashboard** tab provides insight on cloud costs and usage. Select the dashboard to view from the list.
Cost Trend
Trend lines showing the daily costs of the user's resources in each Service Category (or other selected grouping) over the filtered time frame.

Cost Aggregate
Chart showing the division of the user's total resource usage cost between Service Categories (or another selected grouping) over the filtered time frame.

Cloud Tag Values dashboard
Tagged Resources
Set of tagged resources owned by the logged-in user, grouped by resource type.
Assigned Tag Values

Division of assigned tag values among the resources for a selected tag name ("Stack" by default).

**Search the Cloud User Portal**

The Search page returns search results and provides filters that help you to zoom in on the results.

Role required: sn_cmp.cloud_service_user

1. On any portal page, enter text in the **Search** text box and press **Return**.
2. To filter the search results on the **Cloud Search** page, click a category.

   In this example, only **Stacks** that include the search text **s** appear in the **Search results** list:
Launch a stack

Launching a stack of cloud resources means submitting a request for an item in the service catalog. You can launch a stack from the Cloud User Portal overview page.

Role required: sn_cmp.cloud_service_user
When you submit a request for an item in the service catalog, the system typically provisions the stack immediately. Sometimes, your request must be approved first.

If a resource on a stack that you request requires credentials when being accessed via SSH, then you must create an SSH key pair before you can launch the stack. See Create an SSH key and Download an SSH key.

1. Open the request form using either of the following methods:
   - On the overview page, click Launch a Stack. By default, all catalog items in all service categories are listed. Select a Service Category, and then click a catalog item Name.
   - Open the Catalog page. By default, all catalog items in all service categories are listed. Select a Service Category and then click Launch on the catalog item.

2. In each section of the form, enter the values that configure your request. The form fields differ for each catalog offering. Contact your Cloud Management administrator if you have any questions about the fields.

3. Click the section header to move to the next section.
   The remaining quota is updated based on the user group that you select. For mandatory fields, the system selects the first item in the list by default and you can select any value that you need.

4. Click Submit to request the stack.
   - The Activities page opens to the Status tab on the Track > Requests activity. The tab displays the progress and status of the request. The following color codes show the stage of the request:
     - Green: Completed or Approved.
     - Orange: Processing.
     - Red: Rejected, Error, or Canceled.
   - If approval is required, then when the administrator approves your request, the system launches the stack and updates the Request Status.
   - Select another request to view its Request Status and Stack Status.
   - To view a summary of all your pending requests, click the Pending Requests link in the title bar at any time.
<table>
<thead>
<tr>
<th>Activity</th>
<th>Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Track</td>
<td></td>
</tr>
<tr>
<td>Requests</td>
<td></td>
</tr>
<tr>
<td>Change Requests</td>
<td></td>
</tr>
<tr>
<td>Incidents</td>
<td></td>
</tr>
<tr>
<td>Tasks</td>
<td></td>
</tr>
<tr>
<td>Lease Operations</td>
<td></td>
</tr>
<tr>
<td>Business Schedule</td>
<td></td>
</tr>
<tr>
<td>Operations</td>
<td></td>
</tr>
<tr>
<td>Monitor</td>
<td></td>
</tr>
<tr>
<td>Cloud Events</td>
<td></td>
</tr>
<tr>
<td>Manage</td>
<td></td>
</tr>
</tbody>
</table>

**Requests**

- **Sundar_Stack**
  - Request ID: REQ0010522
  - Request Item ID: RITM0010522
  - Created about a month ago

- **Anz Volume**
  - Request ID: REQ0010521
  - Request Item ID: RITM0010521
  - Created about a month ago

- **arm two win**
  - Request ID: REQ0010515
  - Request Item ID: RITM0010515
  - Created about a month ago

**Request Status**

- **Stack Status**
  - **Status**: 2018-10-17 11:20:55 - Could not reach...
Track a stack request
The system generates a request when you submit the form to launch a stack.

Role required: sn_cmp.cloud_service_user
If no approval policy is defined, requests are auto-approved. If a policy is defined, then a user on the approver list must approve the request.

1. Go to the Activities page or navigate to Overview > View Activities, then select Track > Requests.
The following color codes show you the status of the request:
- Green: Completed or Approved.
- Orange: Processing.
- Red: Rejected, Error, or Canceled.

2. In the Requests list, click an entry.

| Status tab | Request Status | Actions taken on the request. Click the down-arrow (▼) to toggle between:
| --- | --- | --- |
|  |  | •Verbose mode that describes each action and the status of the action
|  |  | •Icon-only mode that indicates only the status (pass/fail) of an action.

| Status tab | Stack Status | Status of each stack request. If deployment fails, errors from cloud orchestration are listed.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Comments / Work Notes tab | Item requested and request number | Log of comments and notes made for the request. The itil role is required to add a comment or work note.
To add a comment or work note:
1. Enter text.
2. Select the Work notes check box to add a work note.
3. Click Send.

| Request Details tab | Values that the requester submitted on the stack request form.

| Tasks tab | Lists all the catalog tasks created for stacks that failed to provision. Click on a catalog task to view more details about that task. See Resubmit a failed stack request.

Resubmit a failed stack request
The Cloud Management application creates a remediation or a catalog task when a request for a stack fails to provision. You can resubmit the failed request or assign it to the Cloud Operator group to handle it for you.

Roles required: sn_cmp.cloud_service_user and sn_cmp.cloud_operator.
Task remediation captures every remediation applied to a failed request. You can view the remediation carried out for a failed request and all the major milestones in the Comments/Work Notes sub tab in the Cloud User Portal. Task remediation is available only for new provisioning of stacks and not for life cycle or any other operation.
A stack may fail to provision for many reasons. A stack may fail if you have exceeded your cloud quota, datacenter capacity, or have entered an invalid parameter in the request form (such as a duplicate NIC).

If a stack fails to provision, a catalog task is created and assigned to you. An error message pertaining to the failed request is displayed in the Status sub tab in the Cloud User Portal.

1. In the Cloud User Portal, navigate to **View Activities > Requests > Tasks**.
2. Click the catalog task to resubmit.
3. Based on your preference, select an action:
   - **Retry**: Resubmits the failed request. You can try to resubmit the failed request by resolving the error if possible. If the error is due to exceeding the quota limit, you can try to deprovision some resources that are not currently in use and then click **Retry**. When you click **Retry**, the original request is resubmitted. A new request is not created.
     - **Note**: Any policy associated with a resource operation is re-executed.
   - **Ask for Help**: Creates a new catalog task and assigns it to the Cloud Operator group to resubmit the failed request. The Cloud Operator group can look more closely into the error and try to resolve it for you. Every time the comments or work notes in the catalog task gets updated by the Cloud Operator group, the changes are reflected in the **Comments/Work Notes** sub tab in the Cloud User Portal. If necessary, the operator can cancel the failed request.
   - **Cancel Order**: Cancels the failed request and deprovisions all partially-provisioned resources.

Manage a stack

Use the Stack Details page to view details and status for a stack and to perform life cycle operations on a stack.

Role required: sn_cmp.cloud_service_user

1. Use one of the following methods to open the **Stack Details** page:
   - Click **Stacks** on the toolbar.
   - Use the **Search** text box to search for the stack.
   - On the home page, click a stack in the **Recent Stacks**, or the **Stack Health** section.
   - On the home page, click **Manage Stacks**.

By default, all stacks in all service categories are listed.

2. In the **Stacks** list, select a stack category, and then click a stack **Name**.
   The **Stack Details** page displays the following information:
• Breadcrumb navigation
Name of the stack. Click View Dependency to view the dependency map for the stack, Manage Stack to view the resources, properties, and activities of the stack, and the Track Operations to view the list of lifecycle operations performed on the stack and its resources.

Status information
List of all resources in the stack. Click a resource to view details in the Properties section. Color codes indicate status of the stack:
- Green: On/Active.
- Yellow: Turned off or Processing.
- Red: Terminated or Error.

List of the stack properties. Some properties are set in the stack request form and others are set by policy.

Activities associated with the stack. Click a tab to view:
- Change requests that are associated with the stack. See Track a change request.
- Incidents that were raised for the stack. See Submit an incident for a stack and Track an incident.
- Cloud events that are associated with the stack. Cloud Events appears only when a resource is selected. See View cloud events.
- Catalog tasks associated with the stack. See Resubmit a failed stack request.
- Upcoming business hours schedule operations associated with the stack. You can skip an upcoming operation or cancel a schedule associated with the stack. See Modify business schedule operations for a stack.

Operations you can perform on a stack. See Perform a life-cycle operation on a stack or resource.

Text box used to search the Cloud User Portal for stacks, resources based on types, stack requests, change requests, incidents, keys, and catalog items.

### Stack properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name properties</td>
<td>Name that you specified when requesting the stack.</td>
</tr>
<tr>
<td>Stack Status</td>
<td>Current status of the stack:</td>
</tr>
<tr>
<td></td>
<td>• Active (provisioned stacks)</td>
</tr>
<tr>
<td></td>
<td>• Terminated (deprovisioned stacks)</td>
</tr>
<tr>
<td></td>
<td>• Unmanaged (discovered stacks)</td>
</tr>
<tr>
<td></td>
<td>• Error (errored stacks)</td>
</tr>
<tr>
<td></td>
<td>• Processing (stacks being processed)</td>
</tr>
<tr>
<td>Assigned To</td>
<td>Owner of the stack configuration.</td>
</tr>
<tr>
<td>Cloud Account</td>
<td>Cloud account that the stack is associated with.</td>
</tr>
<tr>
<td>Service Request Item</td>
<td>Record created in Request Item table when you ordered the catalog which resulted in this stack.</td>
</tr>
<tr>
<td>Blueprint</td>
<td>Blueprint with which the stack is associated.</td>
</tr>
<tr>
<td>Owned by</td>
<td>User name of the requester.</td>
</tr>
<tr>
<td>Schedule</td>
<td>Mentions if a schedule is associated with the stack.</td>
</tr>
<tr>
<td>Output</td>
<td>Displays details specific to ARM/CFT stacks, such as the URL of an application, names or locations of any resources created.</td>
</tr>
<tr>
<td>Updated</td>
<td>Date and time the stack details were last updated.</td>
</tr>
<tr>
<td>Created</td>
<td>Date and time the stack was requested.</td>
</tr>
</tbody>
</table>
Manage a resource
Use the Resource Details page to view details and status for a resource and to perform life cycle operations on a resource.

Role required: sn_cmp.cloud_service_user

1. Click Resources on the toolbar.
   The Resources page lists all resource types in the left pane.
| Name            | Status | State  | Assigned to        | Assignment group             | Owned by  | CPUs | GBs | Memory (MB) | Network adapters | Object ID         | Terminated On | Terminated | |
|-----------------|--------|--------|--------------------|------------------------------|-----------|------|-----|-------------|------------------|------------------|----------------|-------------|
| AWS Datacenter  |        |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Virtual Machine Instance |    |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Load Balancer   |        |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Cloud Load Balancer |    |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Network         |        |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Cloud Subnet    |        |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Compute Security Group |  |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Internet Gateway |      |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Cloud Network   |        |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| Storage         |        |        |                    |                              |           |      |     |             |                  |                  |                |             | |
| 1               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 1    | 1   | 627         |                  | 03b7e6c60087760c6 | False          | False      | |
| 2               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 1    | 1   | 627         |                  | 03b7e6c60087760c6 | False          | False      | |
| 3               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 0    | 1   | 628         |                  | 03b7e6c60087760c6 | False          | False      | |
| 4               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 1    | 1   | 627         |                  | 03b7e6c60087760c6 | False          | False      | |
| 5               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 0    | 1   | 628         |                  | 03b7e6c60087760c6 | False          | False      | |
| 6               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 1    | 1   | 627         |                  | 03b7e6c60087760c6 | False          | False      | |
| 7               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 0    | 1   | 628         |                  | 03b7e6c60087760c6 | False          | False      | |
| 8               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 1    | 1   | 627         |                  | 03b7e6c60087760c6 | False          | False      | |
| 9               | Terminated |        | System Administrator | Team Development Code Reviewers |           | 0    | 1   | 628         |                  | 03b7e6c60087760c6 | False          | False      | |
| 10              | Terminated |        | System Administrator | Team Development Code Reviewers |           | 1    | 1   | 627         |                  | 03b7e6c60087760c6 | False          | False      | |
2. Navigate to the appropriate resource type to see the list of resources that you are interested in and then click a resource name to view specific details about that resource.
The resource details screen contains the following information:

- Breadcrumb navigation
- All properties specific to the resource
- Activities associated with the resource. Click a tab to view:
  - Change requests that are associated with the resource. See Track a change request.
  - Incidents that were raised for the resource. See Submit an incident for a stack and Track an incident.
  - Cloud events that happened to the resource. See View cloud events.
- Text box used to search the Cloud User Portal for stacks, resources based on types, stack requests, change requests, incidents, keys, and catalog items.

**Perform a life-cycle operation on a stack or resource**

Operations like Start/Stop, Deprovision, and ExecuteScript are called life-cycle operations or day-2 operations. When you request a life-cycle operation on a stack or resource, the system generates a change request. An approval policy specifies either that the change is auto-approved or that a user on the approver list must approve the change.

Life-cycle operations require the user with sn_cmp.cloud_service_user role who owns the resource. A user with the root_admin, admin, or sn_cmp.cloud_admin role can also perform the operation.

- As noted in the descriptions, not all operations are supported for all providers or for all service categories (resource types).
- All other configuration management providers: If a stack or resource requires login, then you must download the associated SSH key before you can access the resource. See Download an SSH key.

1. Select a stack or resource.
2. While viewing the Stack Details page, select the operation from the Select Stack Operation or Select Resource Operation list.
   
   For full descriptions of the operations, see Life-cycle operations in Cloud Management.
3. Some operations require settings (for example, the name of a script or new schedule settings). Enter the settings and then click OK.
   
   The system submits a change request and the Activities > Change Requests page displays the progress and status of the request.

**Life-cycle operations in Cloud Management**

Operations like Start/Stop, Deprovision, and ExecuteScript are called life-cycle operations or day-2 operations. When you request a life-cycle operation on a stack or resource, the system generates a change request. An approval policy specifies either that the change is auto-approved or that a user on the approver list must approve the change.

To perform an operation, you select a stack or resource and then click Select Stack Operation or Select Resource Operation. The list includes only appropriate operations for the selected resource or stack. Some operations are provided by configuration management systems like Ansible or Terraform.

**Operations in the base system**

**Start**

The Start operation starts selected VM. The resource status changes to On.
No setting is required.

**Stop**

The Stop operation stops the selected VM. The resource status changes to Off.

No setting is required.

**Deprovision**

The Deprovision operation terminates the selected stack and sends a notification to the stack owner.

No setting is required.

**Virtual Machine 'Store Extension Interface' operations**

The Virtual Machine Store Extension Interface operations are supported by Google Cloud Platform and IBM Cloud only. The Terraform Connector ServiceNow Store app supplies the operations.

**Virtual Machine Store Extension Interface.ExecuteScript**

The ExecuteScript operation runs a script on the selected Windows or Linux VM. For example, a script might install an application on a resource. See KB0756431 for detailed instructions.

*Note: To run the ExecuteScript operation, a VM must include credentials. Linux VMs inherit management key credentials when they are provisioned. For Windows VMs, use the InfuseKey operation to add credentials to the VM.*

In the popup, specify:

- **ScriptType**: Shell and PowerShell scripts are supported.
- **Script**: Name of the script.

**Virtual Machine Store Extension Interface.InfuseKey**

For Windows VMs, the InfuseKey operation adds key credentials (public and private key pair values) to the VM.

As a result, you can view the credentials on the Properties list of the Windows VM and can use the credentials to perform VM operations. For instructions on generating a key, see Create an SSH key.

To add key credentials to a Windows VM:

1. Select the Windows VM.
2. In the Select Resource Operation list, select Virtual Machine Store Extension Interface.InfuseKey.
3. In the popup, specify the key to use in the UserKey field and then click OK.

**Virtual Machine Store Extension Interface.Execute Job Template**

Ansible supplies the Execute Job Template operation. The operation enables you to attach a node to an Ansible server and execute a package on that VME. For example, a script might install an application on a resource. In the popup, specify:

- **WorkloadConfigProviderType**
<table>
<thead>
<tr>
<th>WorkloadConfigProvider</th>
<th>Inventory</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostGroup</td>
<td>The list includes all possible applications (called <em>groups</em> in Ansible), not just the applications that the configuration management provider supports. Therefore, if users select an unsupported application, the provisioning fails.</td>
</tr>
<tr>
<td>ApplicationProfile</td>
<td>Select the application profile to execute on the selected resource.</td>
</tr>
</tbody>
</table>

- **ScriptType**: For Google Cloud Platform, **Shell** is supported.
- **Script**: Name of the script.

**Virtual Machine Store Extension Interface: Resize**

The *Resize* operation changes the *Size* setting for a VM (called **HardwareType** in the CI definition). In the popup, specify the new size in the **HardwareType** field.
‘Compute Security Group Store Extension Interface’ operations

The Compute Security Group Store Extension Interface operations are supported for IBM Cloud only. The Terraform ServiceNow Store app supplies the operations.

Compute Security Group Store Extension Interface.Deprovision

The Deprovision operation terminates the selected compute security group resource and sends a notification to the owner.

No setting is required.

The Status of the network interface changes to Terminated. In the Stack Health section of the home page, the status of the stack indicates that some resources are down.
'Load Balancer Store Extension Interface' operations

**Load Balancer Store Extension Interface.Deprovision**

The **Deprovision** operation terminates the selected load balancer and sends a notification to the stack owner.

No setting is required.

The **Status** of the load balancer changes to **Terminated**. In the Stack Health section of the home page, the status of the stack indicates that some resources are down.

'Network Store Extension Interface' operations

'Network Store Extension Interface.Deprovision'

The **Deprovision** operation terminates the selected network store and sends a notification to the stack owner.

No setting is required.

The **Status** of the network store changes to **Terminated**. In the Stack Health section of the home page, the status of the stack indicates that some resources are down.

'Storage Volume Store Extension Interface' operations

The **Storage Volume Store Extension Interface** operations are supported for IBM Cloud only. The Terraform ServiceNow Store app supplies the operations.

**Storage Volume Store Extension Interface.Create Snapshot**

The **Create Snapshot** operation generates a read-only copy of the specified volume.

In the popup, specify the **SnapshotName** to use as the label for the snapshot.

The snapshot is saved in the **(cmdbc.ci_storage_vol_snapshot_list)** table.

If approval is not required, the system immediately takes the snapshot. If approval is required, there may be a delay while an admin approves the change. You receive email when the admin approves. After the admin approves, return to the VM Instance form and click the **Proceed with Change** related link.

**Note:** If the VM is running, the VM is shut down at the start of the snapshot process and then restarted when the process finishes. The system auto-deletes the oldest snapshot when the **Snapshot eviction policy** limit is exceeded.

**Storage Volume Store Extension Interface.Restore From Snapshot**

The **Restore From Snapshot** operation restores the image from the specified snapshot.

Select the **SnapshotID** to use to restore the volume.

**Storage Volume Store Extension Interface.Delete Snapshot**

The **Delete Snapshot** operation deletes the selected snapshot.

**Storage Volume Store Extension Interface.Attach**

The **Attach** operation attaches a VM to the selected volume.
1. Select the volume to attach. The **Status** of the volume must be **Available**.

2. Select the **Storage Volume Store Extension Interface.Attach** operation.

3. In the **NodeId** field on the popup, select the **Instance ID** of the server to attach to the volume.

   The status of the volume changes from Available to In Use.

---

**Storage Volume Store Extension Interface.Detach**

The **Detach** operation detaches a VM from the selected volume.

1. Select the volume to attach. The **Status** of the volume must be **In Use**.

2. Select the **Storage Volume Store Extension Interface.Detach** operation.

3. In the **NodeId** field on the popup, select the **Instance ID** of the host to detach from the volume.

   If this VM is the last or the only VM that is detached, then the status of the volume changes from In Use to Available.

---

**Storage Volume Store Extension Interface.Deprovision**

The **Deprovision** operation terminates the selected volume and sends a notification to the stack owner.

No setting is required.

- The billing item for the storage resource is removed.
- The **Status** of the volume changes to **Terminated**. In the Stack Health section of the home page, the status of the stack indicates that some resources are down.
- For IBM Cloud: The storage volume is available in the IBM Cloud Console for an extended period to enable the user to reclaim the storage volume.

---

**Submit an incident for a stack**

Submit an incident for a stack when it is not working as expected or you have questions. When the user responds or solves the issue, the incident is closed.

**Role required:** sn_cmp.cloud_service_user

You can submit an incident for a stack or for an individual resource.

You can also submit an incident through the **Incident application**. Make sure that the **Category** field is set to the value **Cloud Management**. The configuration item should be the stack name against which the incident should be raised and the caller name should be the name of the person for whom this incident is created.

1. View the stack details page using either of the following methods:

   - On the **Stack Details** page, click the **Incidents** tab in the **Activities** section.
   - On the **Activities** page, click **Track > Incidents**.

   The **Activities** section lists all submitted incidents for the stack.

---

| Activities # Incidents | View and submit incidents of all types, for any stack or resource. |
2. Click **New Incident** and fill in the form:

<table>
<thead>
<tr>
<th>Incident type</th>
<th>Type that best matches your need.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stack / Resource (related fields)</td>
<td>Stack and resource that the incident applies to.</td>
</tr>
<tr>
<td></td>
<td>From the Activities page: Select the Stack from the list of active stacks. Leave the Resource field blank if the issue is not associated with a particular resource.</td>
</tr>
<tr>
<td></td>
<td>From the Stack Details page: The Stack value is the current stack.</td>
</tr>
<tr>
<td></td>
<td>From the Resource page: The Resource value is the current resource.</td>
</tr>
<tr>
<td>Describe the issue</td>
<td>Describe the issue that you are experiencing or enter your question. Give full information on the behavior that is the issue.</td>
</tr>
<tr>
<td>What is the business impact</td>
<td>Specify how the incident affects your work and the work of others, including your customers.</td>
</tr>
<tr>
<td>Attach screenshots, logs, and so on</td>
<td>Click or drop to add files that clarify the issue or question.</td>
</tr>
</tbody>
</table>

3. Click **Submit**.

The system generates the incident. The **Incidents** tab on the **Activities** page displays a summary.

To view a list of all submitted incidents, navigate to **Activities > Incidents**.

**Using the Activities page**

Use the Activities page to view and manage details of your cloud actions.
Tasks
Catalog tasks created when a request for a stack fails to provision. See Resubmit a failed stack request.

Lease Operations
View a list of all your stacks that are nearing their lease end dates. You can extend the end date of a lease for any of the stacks listed. See Modify the lease for a stack.

Business Schedule Operations
View a list of upcoming business hours schedule operations associated with stacks. You can skip an upcoming operation or cancel a schedule associated with a particular stack. See Modify business schedule operations for a stack.

Monitor
Cloud Events
List of life cycle events that affect resources assigned to you or to a user in your group. See View cloud events.

Manage
Keys
View existing SSH keys that are assigned to you or generate a key. See Create an SSH key.

Track a change request
The system generates a change request when you perform a life cycle operation on a stack or resource.

Role required: sn_cmp.cloud_service_user

Each change request is either approved by default or, if an approval policy is in place, is approved by a user that is on the approver list.

1. Go to the Activities page or navigate to Overview > View Activities, then select Track > Change Requests.

<table>
<thead>
<tr>
<th>Number</th>
<th>Unique auto-generated number.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Config item</td>
<td>The affected CI.</td>
</tr>
<tr>
<td>Category</td>
<td>Cloud Management</td>
</tr>
<tr>
<td>Short description</td>
<td>Text that was entered by the requester.</td>
</tr>
<tr>
<td>Approval</td>
<td>Status.</td>
</tr>
</tbody>
</table>

2. In the Change Requests list, click an entry to view full details.

Track an incident
Before an incident is resolved and closed, you can view the state of the incident.

Role required: sn_cmp.cloud_service_user

1. View the Incidents page using either of the following methods:
   - On the Activities page, click Track > Incidents.
   - On the Stack Details page, click the Incidents tab in the Activities section.

   The Activities section lists all submitted incidents for the stack.

2. In the Incidents list, click an entry to view details.

<table>
<thead>
<tr>
<th>Number</th>
<th>Unique auto-generated number.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Cloud Management.</td>
</tr>
</tbody>
</table>
View cloud events
You can view the events that are generated from your cloud resources if your administrator configured Cloud Management to monitor them. All events are listed on the Cloud Events tab on the stack details page.

Role required: sn_cmp.cloud_service_user

1. View the cloud events page using either of the following methods:
   - On the Activities page, click Monitor > Cloud Events.
   - On the Stack Details page, click the Cloud Events tab in the Activities section.

The Cloud Events section lists all events for the stack.

<table>
<thead>
<tr>
<th>Created</th>
<th>Timestamp of the arrival of the event.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Source of the event:</td>
</tr>
<tr>
<td>Event name</td>
<td>Name that is provided by Amazon, Azure, or VMware.</td>
</tr>
<tr>
<td>Subject</td>
<td>Text that describes the event.</td>
</tr>
<tr>
<td>Event time</td>
<td>Timestamp of the event.</td>
</tr>
<tr>
<td>Resource ID</td>
<td>Unique ID of the resource that received the life cycle state change or configuration change event.</td>
</tr>
</tbody>
</table>

2. Click an entry to view full details.

Create an SSH key
A stack might include a virtual machine that requires an SSH key for access. To request such a stack, you can either use a key pair that is assigned to you or you can generate another key.
Role required: sn_cmp.cloud_service_user

1. On the Cloud User Portal, click Activities.
2. In the Activities list, select Manage > Keys.
3. In the SSH Key Pair list, click Generate Key Pair, specify a Name for the key, and then click OK.

The system generates the key pair with the Name that you specified.

Download an SSH key
If a resource on a stack requires login, then you must download the associated SSH key before you can access the resource.

Role required: sn_cmp.cloud_service_user

1. On the Cloud User Portal, click Activities.
2. In the Activities list, select Manage > Keys.
3. Click a key in the SSH Key Pair list and then click Download Private Key in the SSH private key section.

Business hour scheduling
You can set up business hour scheduling on all virtual machines in a stack. Business hour scheduling can be set up on stacks at the time of provisioning or on existing stacks.

Schedules can be set for when a stack should be started, stopped, and/or deprovisioned by defining parameters such as business hours, holidays, long weekends.

The administrator can create schedule profiles in the Cloud Admin Portal. See Resource profiles.

While provisioning a stack in the Cloud User Portal, use the Schedule Profile list to select a profile that has already been created by the administrator. Select a time zone in the Schedule Time Zone list which defaults to your system's time zone and indicates the time zone in which the schedule's business hour time frames is enforced.

You can also define the lease end dates for a stack during provision. The lease end date for a stack can be modified at any time. See Modify the lease for a stack.

Once you provision a stack with a schedule and/or lease dates, a series of operations (start, stop, deprovision) occur at the appropriate time set by you. You can skip a business schedule operation or cancel the business schedule operations associated with a stack. See Modify business schedule operations for a stack.

Modify the lease for a stack
When a stack reaches the end of its lease, the system notifies the stack owner and deprovisions the stack. You can modify the lease for a stack before it approaches its end date.

Role required: sn_cmp.cloud_service_user

A lease end date is assigned to all provisioned stacks. A built-in policy titled Lease End ServiceNow with an on Lease end policy trigger has rules defining notifications and operations that are performed before, at, and after the end date of the lease. Lease dates can be modified for active stacks. The system sends a notification to the stack owner one day prior to the end of the lease. On the day the lease ends, the system stops the stack and sends another notification. Seven days after the lease has ended, the stack is terminated and a final notification is sent to the stack owner. You can modify the actions taken and the timings of these actions by modifying the Lease End ServiceNow policy. You can modify the default lease duration for all new stacks, going forward, by modifying the associated action script in the Scheduling Defaults built-in policy.

1. Navigate to Overview > View Activities, then select Lease Operations.
   A list of stacks with upcoming lease-related operations is displayed.
2. Click the Select Action list associated with the stack and then select Modify Lease.
   The Modify Lease option is also available from the stack details list in Manage Stacks.
3. Set a new date and time for the lease to expire and then click OK.

Modify business schedule operations for a stack
Business hours scheduling can be performed on stacks at the time of provisioning or on existing stacks.

Role required: sn_cmp.cloud_service_user.

Once a stack is set up with a business schedule, the system generates a series of tasks for starting and stopping the stack. You can view these tasks and can modify the business schedule for any stack.

1. Navigate to Overview > View Activities, then select Business Schedule Operations. A list of stacks with upcoming business schedule operations is displayed.
2. From the Select Action list associated with the stack, perform either of the two operations:
   - Skip Operation- Click to skip a single business schedule operation on a stack. A confirmation message appears. Click OK.
   - Cancel Schedule- Click to cancel business hour scheduling on a stack. A confirmation message appears. Click OK.

For modifying the schedule of an existing stack, navigate to the stacks details list in Manage Stacks and select Modify Schedule from the Select Stack Operation list.

Manage personal and group resources

You can view and manage your resources as well as the resources assigned to the group you belong to. If you belong to more than one group, you can view and manage all resources assigned to those groups.

Role required: sn_cmp.cloud_group_admin

1. On the Cloud User Portal, click the Views filter.
2. Select an option:
   - Mine: View and manage only resources that are assigned to you.
   - My Groups: View and manage all resources that are assigned to you and that are assigned to all members of all groups that you belong to.

Only the Root Admin and the Sys Admin have access to all three filters: Mine, My Groups, and All.

Cloud Management troubleshooting

There are several methods to troubleshoot issues with Discovery and the management of cloud resources.

Schemas of Cloud Management tables

The tables are cloud-agnostic and can therefore hold data for any cloud provider.

Load balancer table

The load balancer table (cmdb_ci_cloud_load_balancer) extends from (cmdb_ci_vm_object), which extends from (cmdb_ci). Click to enlarge the image.
Key native attributes of the cmdb_ci_cloud_load_balancer table

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object_id</td>
<td>Identifier that typically holds the load balancer name as the value.</td>
</tr>
<tr>
<td>canonical_hosted_zone_id</td>
<td>ID of the Amazon Route 53 hosted zone for the load balancer.</td>
</tr>
<tr>
<td>canonical_hosted_zone_name</td>
<td>DNS name of the load balancer.</td>
</tr>
<tr>
<td>dns_name</td>
<td>Public DNS name of the load balancer.</td>
</tr>
<tr>
<td>fqdn</td>
<td>DNS name as fully qualified domain name. Can also be a CNAME record pointed to the DNS name.</td>
</tr>
<tr>
<td>state</td>
<td>State of the load balancer: available or terminated.</td>
</tr>
</tbody>
</table>

Network resource table

The network resource table (cmdb_ci_network) extends from (cmdb_ci_vcenter_object), which extends from (cmdb_ci_vm_object), which extends from (cmdb_ci). Click to enlarge the image.
**Key native attributes of the cmdb_ci_network table**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object_id</td>
<td>Identifier that typically holds the network ID as the value. Uniquely identifies an object within a cloud.</td>
</tr>
<tr>
<td>name</td>
<td>Name of the network.</td>
</tr>
<tr>
<td>cidr</td>
<td>IP Address range. Classless inter-domain routing is a set of internet protocol standards that is used to create unique identifiers for networks.</td>
</tr>
<tr>
<td>default_gateway</td>
<td>Holds InternetGatewayID if we attach InternetGateway to the network. A default gateway serves as an access point or IP router that a networked computer uses to send information to a computer in another network or Internet. The specified gateway is used by default unless an application specifies a different gateway.</td>
</tr>
<tr>
<td>Broadcast_address</td>
<td>IP address used to transmit messages and data packets to network systems.</td>
</tr>
<tr>
<td>Is_shared</td>
<td>Boolean. true: Network shared across other projects. false: Network not shared across other projects.</td>
</tr>
<tr>
<td>max_ports</td>
<td>Maximum number of hosts that can be connected to the network.</td>
</tr>
<tr>
<td>Is_external</td>
<td>Boolean. true: Network is external. false: Network is internal.</td>
</tr>
<tr>
<td>terminated_on</td>
<td>Time that the network was de-provisioned.</td>
</tr>
<tr>
<td>state</td>
<td>State of the network: <strong>available</strong> or <strong>terminated</strong>.</td>
</tr>
<tr>
<td>netmask</td>
<td>Type of CIDR. 32-bit mask that divides an IP address into subnets and specify the hosts that are available on the network.</td>
</tr>
<tr>
<td>dhcp_enabled</td>
<td>Boolean. true: Dynamic IP address is assigned to host false: Static IP address is assigned to host</td>
</tr>
</tbody>
</table>

**Storage volume resource table**

The storage volume resource table (cmdb_ci_storage_volume) extends from (cmdb_ci). Click to enlarge the image.
Key native attributes of the cmdb_ci_storage_volume table

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object_id</td>
<td>Identifier that typically holds the object ID as the value. Uniquely identifies an object within a cloud.</td>
</tr>
<tr>
<td>volume_id</td>
<td>Identifier that typically holds the volume ID as the value. Uniquely identifies an object within a cloud.</td>
</tr>
<tr>
<td>volume_container</td>
<td>For NetApp only, LUN becomes the volume and NetApp volume becomes the volume container.</td>
</tr>
<tr>
<td>sharable</td>
<td>Boolean. true: Volume is shared by multiple VMs. false: Volume is not shared by multiple VMs.</td>
</tr>
<tr>
<td>storage_type</td>
<td>Type of storage. • AWS: Block • Azure: PageBlob • vSphere: VMware vdisk</td>
</tr>
<tr>
<td>size</td>
<td>Total capacity of the volume.</td>
</tr>
<tr>
<td>free_space</td>
<td>Available space of the volume</td>
</tr>
<tr>
<td>state</td>
<td>State of the volume: available or in_use.</td>
</tr>
<tr>
<td>share_count</td>
<td>Number of VMs that are shared by the volume.</td>
</tr>
</tbody>
</table>

Virtual server resource table

The virtual server resource table (cmdb_ci_vm_instance) extends from (cmdb_ci_vm_object), which extends from (cmdb_ci). Click to enlarge the image.
Key native attributes of the cmdb_ci_vm_instance table

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object_id</td>
<td>Identifier that typically usually holds the VM instance ID. Uniquely identifies an object within a cloud.</td>
</tr>
<tr>
<td>name</td>
<td>Name of the VM.</td>
</tr>
<tr>
<td>state</td>
<td>State of the VML: on, off, or terminated.</td>
</tr>
<tr>
<td>cpus</td>
<td>Number of CPUs.</td>
</tr>
<tr>
<td>memory</td>
<td>Memory size in megabytes.</td>
</tr>
<tr>
<td>disks</td>
<td>Number of disk drives.</td>
</tr>
<tr>
<td>disk_size</td>
<td>Total size of disks in gigabytes.</td>
</tr>
<tr>
<td>nics</td>
<td>Number of network interface adapters.</td>
</tr>
<tr>
<td>terminated_on</td>
<td>Time that the instance was terminated.</td>
</tr>
<tr>
<td>termination_protection</td>
<td>Boolean. Default value is false. Can prevent the instance from being accidentally terminated using Amazon EC2. false: Cannot prevent the instance from being accidentally terminated using Amazon EC2.</td>
</tr>
</tbody>
</table>

Cloud Optimization: Cloud Insights application

The ServiceNow® Cloud Optimization: Cloud Insights application, available in the ServiceNow Store, helps you to analyze the full range of costs associated with cloud assets so you can identify and take action on opportunities to save money and optimize operations. Release 1.0 ad 1.1 support Amazon Web Services (AWS) only. Release 1.2 also supports Azure.S

Request apps on the Store

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store. For cumulative release notes information for all released apps, refer to the ServiceNow Store version history release notes.

Explore
- Visit the ServiceNow Store website to read the release notes for this new app.

Set up
- Cloud Insights setup guide
- Configure Business Hours operations
- Define or update a Business Hours policy
- Configure Rightsizing operations
- Schedule a resource to be rightsized
- Configure Unassigned Resources operations
- Define or update an Unassigned Resources policy

Administer
- Cloud Insights admin guide
- Spend optimization: Overall spend reports
- Spend optimization: Business Hours reports
- Cost optimization: Rightsizing reports
- Monitoring executions of download and reporting jobs
- Viewing the overall state of business hours usage
- Viewing the overall state of Rightsizing efforts
Use
- Business Hours: Ensuring resources run only when required
- Rightsizing: Implementing cost-effective sizing for resources
- Identifying unassigned resources to ensure accurate reporting

Develop
- Developer training
- Developer documentation

Integrate
- Installed with pages

Troubleshoot and get help
- Ask or answer questions in the Cloud Insights forum
- Search the HI Knowledge Base for known error articles
- Contact ServiceNow Technical Support

Upgrade Cloud Insights from release 1.0 to release 1.1

You upgrade Cloud Insights by upgrading each required app. Cloud Insights release 1.1 is supported on New York and Orlando platforms.

Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to System Applications > All Available Applications > All.
2. In the search box, enter *cloud.
3. Find Cloud Integrations Core.
   The selection list displays 1.0 (Installed).
4. In the selection list, select 1.1 and then click Upgrade.
5. Repeat the process to perform the upgrade for each application in exactly the following order:
   - Cloud Integrations AWS
   - Cloud Spend Reports Core
   - Cloud Spend Reports AWS
   - Cloud Insights Core
   - Cloud Insights AWS
   - Cloud Insights

6. Orlando instances only: Install the sn_itom_pattern application.

Cloud Insights release 1.1 is ready to use.

Cloud Insights setup guide
The Cloud Insights app is available on the ServiceNow Store.
Requirements and limitations

- You must have AWS Management Console administrator permissions to work in the AWS Management Console.
- The Cloud Insights app operates entirely independently of the Cloud Management application. The Cloud Management application is available as a separate subscription and requires the Cloud Management plugin (com.snc.cloud.mgmt).
- Cloud Insights is not supported on mobile devices.
- The Unassigned Resources, Business Hours Reports, and Rightsizing Recommendations features support only Amazon Elastic Compute Cloud (EC2) virtual machine (VM) instances.
- Values in reports might vary slightly from provider billing values due to currency conversion or rounding.
- Domain separation is not supported in this application. For more information, see Application support for domain separation.
## Setting up Cloud Insights

1. **Get the app.**
   - Get the Cloud Insights app from the ServiceNow Store.
   - **Do this:** Visit the ServiceNow Store website to get the Cloud Insights app and supporting apps.

2. **Activate plugins.**
3. **Assign roles.**
4. **Create credentials.**
5. **Configure MID Servers.**
6. **Discover your resources.**
7. **Schedule download jobs.**
8. **Set up insights features.**

---

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Do this</th>
</tr>
</thead>
</table>
| ![Activate all supporting plugins and applications.](image) | Activate all supporting plugins and applications. | For instructions, see:  
  - Request a plugin  
  - Activate a plugin |

When you first open the app, the home page shows the areas to configure. (After you configure the system, the page shows overview data and gives quick access to reports.)

Cloud Insights provides visibility, management, and governance for your cloud infrastructure and your business.

Follow the instructions in the table. The buttons on the Home page provide shortcuts to some setup activities:

- When you click the **Set up Discover Schedule** button in the Configure Accounts and Discovery section, the Discover Schedules form opens.
- When you click the **Set up Download Schedule** button in the Configure Billing section, the Billing Download Jobs form opens.
- When you click the **Set up Download Schedule** button in the Configure Price Sheets section, the Price Sheets Download Jobs form opens.
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>![People icon]</td>
<td>Assign roles to Cloud Insights users and groups.</td>
<td>Roles for Cloud Insights users</td>
</tr>
<tr>
<td>![Checkmark icon]</td>
<td>Create the credentials that enable Cloud Insights to access your AWS data.</td>
<td>Create the credentials that enable Cloud Insights to access your AWS data</td>
</tr>
<tr>
<td>![Server icon]</td>
<td>Configuring MID Servers to connect Cloud Insights to AWS accounts.</td>
<td>Configuring MID Servers to connect Cloud Insights to AWS accounts</td>
</tr>
<tr>
<td>![Disco icon]</td>
<td>Discovering your AWS cloud resources.</td>
<td>Discovering your AWS cloud resources</td>
</tr>
<tr>
<td>![Download icon]</td>
<td>Schedule and manage the jobs that download AWS billing data.</td>
<td>Schedule and manage the jobs that download AWS billing data</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Checkmark]</td>
<td>Schedule and manage the jobs that download price sheets.</td>
<td></td>
</tr>
</tbody>
</table>

When you click the **Set up Download Schedule** button in the Configure Price Sheets section, the Price Sheets Download Jobs form opens. A Price Sheet Download job downloads and stores price sheet data. Rightsizing processes use price sheet data when generating rightsizing recommendations.

Configure the insights features:
- Rightsizing
- Business Hours
- Unassigned Resources

- A Business Hours job applies Business Hours policies to identify resources that are running when they should be powered off, reports them, and can start and stop them on the schedule that you specify. Running only during specified business hours can significantly reduce your cloud spend.
  - The Rightsizing feature analyzes resource usage to recommend better sizes for resources that are wasting money by being over-provisioned or underused. A confidence rating and predicted savings support each recommendation to rightszie a resource.
  - Unassigned Resources policies help you to identify the resources that are not associated with an assignment group and to assign them appropriately. When a resource is assigned to the correct group, the resource can be appropriately governed even as it goes through stages such as patching, upgrading, and reconfiguring.

- **Business Hours**: Ensuring resources run only when required
- **Rightsizing**: Implementing cost-effective sizing for resources
- **Identifying unassigned resources to ensure accurate reporting**

**Roles for Cloud Insights users**

You assign Cloud Insights roles to user groups and to individual users based on user activities and responsibilities.
## Role relationships in Cloud Insights

![Diagram showing role relationships in Cloud Insights]

<table>
<thead>
<tr>
<th>Role title (name)</th>
<th>Tasks and access rights</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cloud Insights Admin</strong>&lt;br&gt;(sn_clin_core.insights_admin)</td>
<td>The role is in the Cloud Insights Core plugin. Typically the person who is financially responsible.&lt;br&gt;• All rights of all other Cloud Insights roles.&lt;br&gt;• All rights of the PA Admin role.&lt;br&gt;• All rights of the (schedule_admin) role.&lt;br&gt;• All rights of the (itil) role.&lt;br&gt;• Define Business Hours and Unassigned Resources policies.&lt;br&gt;• View spend optimization reports.&lt;br&gt;• Add report extensions.&lt;br&gt;• Perform the actions that Cloud Insights recommends.</td>
</tr>
<tr>
<td><strong>Cloud Spend Admin</strong>&lt;br&gt;(sn_cld_spend_core.spend_admin)</td>
<td>The role is in the Cloud Spend Reports Core plugin.&lt;br&gt;• Edit the Cloud Billing dashboards.&lt;br&gt;• All rights of the Cloud Integrations Admin role.&lt;br&gt;• All rights of the (sn_capi.cloud_developer) role.</td>
</tr>
<tr>
<td><strong>Cloud Integrations Admin</strong>&lt;br&gt;(sn_cld_intg_core.cloud_integrations_admin)</td>
<td>The role is in the Cloud Integrations Core plugin.&lt;br&gt;• Configure Billing Download jobs and Price Sheet Download jobs.&lt;br&gt;• All rights of the API Integrator role.&lt;br&gt;• All rights of the (flow_operator) role.&lt;br&gt;• All rights of the (discovery_admin) role.</td>
</tr>
</tbody>
</table>
Create the credentials that enable Cloud Insights to access your AWS data

To enable Discovery to access your AWS account, you create a programmatic user in the AWS Management Console. You specify the credentials of the user when you configure the MID Servers that communicate with your cloud provider.

You should be familiar with AWS policies. If you use IAM, you must know how to create an IAM user and set up a user policy. See the AWS documentation on IAM for details.

Roles required:
- On the AWS Management Console: AWS Management Console administrator
- Cloud Insights: admin or sn_cmp.cloud_admin

You create the credentials in the AWS Management Console and then create a record of them in Cloud Insights.

1. On the AWS Management Console, enter IAM in the AWS services search box to open the Identity and Access Managements (IAM) service.
2. On the IAM Resources portal, click Users.
3. Click Add user.
4. On the Details page, configure the user settings as shown and then click Next.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>User name</td>
<td>Name for the programmatic user, for example, servicenowcloud.</td>
</tr>
<tr>
<td>Access type</td>
<td>Programmatic access.</td>
</tr>
</tbody>
</table>

5. On the Permissions page, configure the following settings and then click Next.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set permissions for &lt;username&gt;</td>
<td>Attach existing policies directly.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
ServiceNow    New York    IT Operations Management

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attach one or more policies</td>
<td>The appropriate policy.</td>
</tr>
</tbody>
</table>

**Note:** The AdministratorAccess policy has the most powerful permission level, including permission to provision cloud resources. The policy enables the same access that would be granted to the instance as if you were not using IAM and used your AWS account Access Key ID and Secret Access Key. You might instead prefer to create a policy or combine multiple policies to grant the appropriate permission level. See [Create an IAM user policy for Cloud Insights](#) for details.

6. On the Review page, verify your selections and then click **Create user**.

7. On the Complete page, click **Download .csv** to save a CSV backup file that contains the user name, Access key ID, and the Secret access key value. You create the file as a backup in case you lose those values. Verify that the file was created and then store the file securely.

8. Create a record of the credentials in Cloud Insights.
   a) In Cloud Insights, navigate to **AWS Settings > Credentials**.
   b) On the AWS Credentials form, click **New**, enter a unique and meaningful **Name** (for example, **AWS Billing Credentials**), and then fill in the form.

   **AWS Credentials form fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Input value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a unique and descriptive name for the AWS credentials.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to use the credential.</td>
</tr>
<tr>
<td>Access Key ID</td>
<td>Enter the <strong>Access key ID</strong> that you generated on the AWS Management Console, such as: <code>APIAIOFODNN7EXAMPLE</code>.</td>
</tr>
<tr>
<td>Secret Access Key</td>
<td>Enter the <strong>Secret access key</strong> that you generated on the AWS Management Console, such as: <code>wPalrXUtnFEMI/K7MDENG/bPxfRfiCYEXAMPLEKEY</code>.</td>
</tr>
</tbody>
</table>
   c) Click **Update** or **Submit**.

**Create an IAM user policy for Cloud Insights**

If you manage users with IAM, you must create an IAM user profile that enables Discovery to access AWS data.

You must know how to create an IAM user and set up a user policy. See the [AWS documentation on IAM](#) for details.

Use an auto-generated access key. You need the key information when you configure AWS credentials in the instance.
Roles required:

- On the AWS Management Console: AWS Management Console administrator
- Cloud Insights: admin or sn_cmp.cloud_admin

1. Log in to the AWS Management Console and create a new user in IAM.
2. Save the Access Key ID and Secret Access Key.
3. Open the user record in the instance for appropriate user.
4. Define a user policy in AWS using either of the following methods:

   - Grant Administrator Access to the instance, which is essentially the same access that would be granted to the instance as in the case that you were not using IAM and used your AWS account Access Key ID and Secret Access Key. Attach the AdministratorAccess policy to the user profile.

   - Create a custom policy with a descriptive name and the following code in the Policy Document field in the user policy. Be sure to see the code listed in this topic.

   ```json
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Action": [
           "cloudwatch:GetMetricData",
           "cloudwatch:ListMetrics"
         ],
         "Resource": "*"
       },
       {
         "Action": "cloudfront:*",
         "Effect": "Allow",
         "Resource": "*"
       },
       {
         "Action": "s3:*
         "Effect": "Allow",
         "Resource": "arn:aws:s3:::*"
       },
       {
         "Action": "elasticloadbalancing:*
         "Effect": "Allow",
         "Resource": "*"
       },
       {
         "Action": "sqs:*
         "Effect": "Allow",
         "Resource": "arn:aws:sqs:*"
       },
       {
         "Action": "rds:*
         "Effect": "Allow",
         "Resource": "*
       },
       {
         "Action": "sns:*
         "Effect": "Allow",
         "Resource": "arn:aws:sns:*"
       },
       {
         "Action": "ec2:*
         "Effect": "Allow",
         "Resource": "*"
       },
       {
         "Action": "cloudformation:*
         "Effect": "Allow",
         "Resource": "*"
       },
       {
         "Action": "directconnect:*
         "Effect": "Allow",
         "Resource": "*"
       }
   ]
   ```
"Resource": "*"
}, {
   "Action": "route53:*",
   "Effect": "Allow",
   "Resource": "arn:aws:route53::*"
}, {
   "Action": ["iam:DeleteServerCertificate",
               "iam:GetServerCertificate",
               "iam:ListServerCertificates",
               "iam:UpdateServerCertificate",
               "iam:UploadServerCertificate"],
   "Effect": "Allow",
   "Resource": "arn:aws:iam::*:server-certificate/*"
}, {
   "Version": "2012-10-17",
   "Statement": [{
      "Action": ["rds:DescribeDBInstances"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }, {
      "Action": ["elasticloadbalancing:DescribeLoadBalancers"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }, {
      "Action": ["ec2:DescribeAvailabilityZones",
                  "ec2:DescribeImages",
                  "ec2:DescribeInstances",
                  "ec2:DescribeInternetGateways",
                  "ec2:DescribeKeyPairs",
                  "ec2:DescribeLoadBalancers",
                  "ec2:DescribeNetworkInterfaces",
                  "ec2:DescribeRegions",
                  "ec2:DescribeRouteTables",
                  "ec2:DescribeSecurityGroups",
                  "ec2:DescribeSubnets",
                  "ec2:DescribeVolumes",
                  "ec2:DescribeVpcs",
                  "ec2:DescribeVpnGateways"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }]
}]

Use the following code snippets to activate the billing, cloud watch, forecast, and actions plugin operations.

**Billing**

```json
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "VisualEditor0",
         "Effect": "Allow",
         "Resource": "*"
      }
   ]
}
```
"Effect": "Allow",
"Action": [
"s3:ListBucketByTags",
"s3:GetLifecycleConfiguration",
"s3:GetBucketTagging",
"s3:GetInventoryConfiguration",
"s3:GetObjectVersionTagging",
"s3:ListBucketVersions",
"s3:GetBucketLogging",
"s3:ListBucket",
"s3:GetAccelerateConfiguration",
"s3:GetBucketPolicy",
"s3:GetObjectVersionTorrent",
"s3:GetObjectAcl",
"s3:GetEncryptionConfiguration",
"s3:GetBucketRequestPayment",
"s3:GetObjectVersionAcl",
"s3:GetObjectTagging",
"s3:GetMetricsConfiguration",
"s3:HeadBucket",
"s3:GetBucketPublicAccessBlock",
"s3:GetBucketPolicyStatus",
"s3:ListBucketMultipartUploads",
"s3:GetBucketWebsite",
"s3:ListJobs",
"s3:GetBucketVersioning",
"s3:GetBucketAcl",
"s3:GetBucketNotification",
"s3:GetReplicationConfiguration",
"s3:ListMultipartUploadParts",
"s3:GetObject",
"s3:GetObjectTorrent",
"s3:GetAccountPublicAccessBlock",
"s3:ListAllMyBuckets",
"s3:DescribeJob",
"s3:GetBucketCORS",
"s3:GetAnalyticsConfiguration",
"s3:GetObjectVersionForReplication",
"s3:GetBucketLocation",
"s3:GetObjectVersion"
],
"Resource": "*"
}

Cloud watch

{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Sid": "VisualEditor0",
    "Effect": "Allow",
    "Action": [
      "cloudwatch:GetMetricData",
      "cloudwatch:ListMetrics"
    ],
    "Resource": "*"
  }
  ]
Forecast

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": ["ce:GetCostForecast"],
      "Resource": "*"
    }
  ]
}
```

Actions

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": ["ec2:DescribeInstances", "ec2:StartInstances", "ec2:ModifyInstanceAttribute", "ec2:StopInstances", "ec2:DescribeInstanceStatus"],
      "Resource": "*"
    }
  ]
}
```

Use this policy to grant access to AWS Cost Explorer APIs for the IAM user that you configure to access billing credentials on the Service account

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": ["ce:GetCostAndUsage", "ce:GetCostForecast"],
      "Resource": "*"
    }
  ]
}
```

Configuring MID Servers to connect Cloud Insights to AWS accounts

To ensure secure and reliable communications, the Discovery process communicates with your cloud provider accounts and cloud resources through one or more MID Servers. You can set up the MID Servers on your network or in one of your cloud networks.
Requirements

There must be an internal network connection between the MID Servers and the AWS Cloud API endpoints: *.amazonaws.com.

Using a proxy server

You can use a proxy server for the Cloud Insights MID Server. See Proxy server configuration for your Cloud Insights MID Server.

MID Server settings

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supported Applications</td>
<td>· Cloud Actions</td>
</tr>
<tr>
<td></td>
<td>· Cloud Billing</td>
</tr>
<tr>
<td>Capabilities</td>
<td>· Cloud Actions</td>
</tr>
<tr>
<td></td>
<td>· Cloud Billing</td>
</tr>
<tr>
<td></td>
<td>· AWS</td>
</tr>
</tbody>
</table>

Proxy server configuration for your Cloud Insights MID Server

You can configure any MID Server to use a proxy server for Cloud Insights operations. Proxy servers support all cloud-based activities such as running Discovery, Billing Download jobs, and Price Sheet Download jobs.

Proxy server limitations

- Only Windows or Linux platforms are supported.
- The Google Cloud Platform integration is not supported.
- The VMware integration is not supported.
- Remote PowerShell scripts cannot be executed.
- Custom APIs might not work.

Supported proxy server authentication for Cloud Insights

<table>
<thead>
<tr>
<th>Proxy Server type</th>
<th>Authentication type</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP/HTTPS</td>
<td>No authentication</td>
</tr>
<tr>
<td>SOCKS5</td>
<td>No authentication</td>
</tr>
<tr>
<td>HTTP/HTTPS</td>
<td>Basic authentication</td>
</tr>
<tr>
<td>SOCKS5</td>
<td>Basic authentication</td>
</tr>
<tr>
<td>HTTP/HTTPS</td>
<td>NTLM</td>
</tr>
</tbody>
</table>
Supported Proxy server configurations

Supported configuration settings

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Operating system</th>
<th>Proxy server</th>
<th>Authentication mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration 1</td>
<td>Linux</td>
<td>None</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Configuration 2</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>None</td>
</tr>
<tr>
<td>Configuration 3</td>
<td>Linux</td>
<td>Squid (HTTPS)</td>
<td>Local</td>
</tr>
<tr>
<td>Configuration 4</td>
<td>Windows</td>
<td>Squid (HTTPS)</td>
<td>Active Directory</td>
</tr>
</tbody>
</table>

Supported configurations

<table>
<thead>
<tr>
<th>Cloud provider</th>
<th>Functionality</th>
<th>Configuration 1</th>
<th>Configuration 2</th>
<th>Configuration 3</th>
<th>Configuration 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td>Cloud Discovery</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM Provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>VM life cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>CFT provisioning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Billing Download jobs and Price Sheet Download jobs</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Cloud Insights</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using SSH</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Scripts using PowerShell</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td>Cloud events</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

Discovering your AWS cloud resources

When you create a schedule with the Discovery Manager wizard, you select the service accounts to discover, the credentials for accessing the accounts, and the MID Servers that scan the resources. The wizard greatly simplifies the configuration process for discovering cloud attributes.

Initial set up from the Home page

During initial installation, on the Home page, when you click the Set up Discover Schedule button in the Configure Accounts and Discovery section, the Discover Schedules form opens. Click Cloud Discovery to proceed.
About Cloud Discovery

The Cloud Discovery plugin includes the necessary components from Cloud Management to perform Cloud Discovery. See [Cloud Discovery](#) for detailed instructions for configuring Cloud Discovery.

Cloud Discovery finds the following cloud attributes for access to your AWS account:

- **Service accounts, subaccounts, and logical datacenters (LDCs):** Cloud Discovery finds all logical datacenters and subaccounts in a cloud service account.
- **Cloud resources:** Cloud Discovery by IP ranges finds all virtual machines in a cloud service account. Select a MID Server configured for Cloud Discovery and create the schedule. Cloud Discovery determines the IP addresses to scan based on the IP ranges in your cloud service account. Make sure that you have one or more MID Servers up and running that can access those addresses.
- **Tracking Cloud Discovery results:** The instance displays all results from a Cloud Discovery on both the cloud account page in Cloud Management and on the Cloud Discovery home page. You can drill down into specific schedules and lists of discovered devices, cloud resources, and errors. Sort these results for a closer look and view Cloud Discovery trends for the devices you find. [Error results](#) give you specific suggestions for fixing any problems you encounter.

Schedule and manage the jobs that download AWS billing data

Billing Download jobs download, organize, and store billing data for each payer account. The system analyzes the data to make recommendations for changes in your cloud operations that can lead to cost savings.

- For each provider, run Discovery on each service account.
- Role required: Cloud Spend Admin (sn_cld_spend_core.spend_admin) or Cloud Integrations Admin (sn_cld_intg_core.cloud_integrations_admin)

On the Home page,

**Note:** If the Cloud Management application is installed on your instance: Both Cloud Management and Cloud Insights download billing data. The two download jobs are completely separate processes and they do not interfere with each other.

- To ensure accurate reporting and recommendations, make sure that Discovery runs before the scheduled execution.
- The Billing Download job updates the billing node data table (sn_cld_intg_aws_cost_usage) with the CIs in the CMDB that correspond to each resource ID.
- Each update of billing data triggers the Business Hours and Rightsizing jobs to analyze spend and usage data and to update the actionable recommendations in reports.
- You can create only one Billing Download job for each payer account.
- For AWS accounts, only costs of unblended type are supported.
- When the scheduled time arrives, job execution happens in multiple stages (for example, connect to the provider, download the data, perform post-import sort, and so on). The system logs status and results on the Job Executions dashboard for each stage.
- Billing data is downloaded in reverse chronological order by month. For example, if the range is from March to June, data for June is downloaded first.
- To ensure meaningful results for the first billing data download, the app downloads data for at least 30 days. For example, if today is June 10 and you specify June for download, the system downloads data for both May and June to ensure at least 30 days of data for analysis. You can download data only for the most recent six months.
To optimize performance, the instance downloads only changes to the bill since the most recent download.

1. If you currently have an AWS Cost and Usage Report, review the settings in this step to ensure that the settings on the report meet the requirements for Cloud Insights. On the AWS Management Console, define an AWS Cost and Usage Report, as follows:
   a) Follow the instructions for “Creating an AWS Cost and Usage Report” in the AWS documentation and use the following settings:
      • Select the Include resource IDs check box.
      • Select the Automatically refresh your Cost and Usage Report when check box.
      • Report path prefix: optional
      • Set Time granularity to Hourly.
      • Set Report versioning to Create new report version.
      • Set Compression type to GZIP or ZIP. Cloud Insights does not support Parquet.
   b) Keep track of the following values because you will enter them into a Cloud Insights form in a moment.
      • Report name.
      • Report prefix.
      • Name of the S3 bucket that is the data storage area for the daily detailed billing records on the AWS account.
      • The cost type that your AWS account generates (unblended or blended). You set this value on the Advanced Options section of the AWS Billing and Cost Management console.

2. During initial installation, on the Home page, when you click the Set up Download Schedule button in the Configure Billing section, the Billing Download Jobs form opens. Navigate to Cloud Insights > AWS Settings > Billing Download Jobs.

3. Click New and then fill in the form.

### Billing Download Job form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Meaningful name for the scheduled execution of the Billing Download job.</td>
</tr>
<tr>
<td>Provider</td>
<td>The provider.</td>
</tr>
<tr>
<td>Service account</td>
<td>The AWS payer account.</td>
</tr>
</tbody>
</table>
Bucket

Amazon S3 bucket that is the data storage area for the daily detailed billing records on the AWS account. Ensure that the credentials associated with the service account have access to the S3 bucket from which the bills are downloaded.

To obtain the value:

1. Log in to the AWS Management Console and select **My Billing Dashboard** in the menu.
2. Select **Reports**.
3. Copy the value from the **S3 bucket** column into the **Bucket** field on the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report prefix</td>
<td>The <strong>report-prefix</strong> value that appears in your AWS Cost and Usage Report.</td>
</tr>
<tr>
<td>Report name</td>
<td>The <strong>report-name</strong> value that appears in your AWS Cost and Usage Report.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for activating the job. Only active jobs are executed.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Notify users / Notify groups</td>
<td>The users or groups to notify by email of the status of the job execution (for example, download failure). The system does not send a notification for success. Users or groups with the Billing Admin (sn_cld_intg_core.cloud_billing_admin) role are well suited to handle these issues. To update the email template, navigate to System Notification &gt; Email &gt; Notifications and open the Notify on billing job execution error template. For information on configuring the email, see Create an email template.</td>
</tr>
<tr>
<td>Run</td>
<td>Frequency to execute the job. Default: Daily</td>
</tr>
<tr>
<td>Time</td>
<td>Time of day to execute the job.</td>
</tr>
<tr>
<td>Last successful execution</td>
<td>Read-only field. The timestamp of the most recent execution.</td>
</tr>
<tr>
<td>Current execution status</td>
<td>Read-only field. The status of the execution that is currently running.</td>
</tr>
<tr>
<td>Current execution details</td>
<td>Read-only field. Execution details for the execution that is currently running.</td>
</tr>
</tbody>
</table>

4. Click Test Connection.
   The Test Connection workflow uses the configured settings to attempt to access the provider account, displays a progress pop-up, and then displays a success message or an error message that suggests actions to fix the configured settings.

5. After the connection succeeds, you can execute immediately or save the scheduled job:

<table>
<thead>
<tr>
<th>Choice</th>
<th>Resulting actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click Submit.</td>
<td>Create the schedule for the Billing Download job.</td>
</tr>
<tr>
<td>Click Execute Now.</td>
<td>Execute the Billing Download job immediately. The Billing Download Job dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>1. In the Start month field, enter the first month of data to download.</td>
</tr>
<tr>
<td></td>
<td>2. In the End month field, enter the last month of data to download.</td>
</tr>
<tr>
<td></td>
<td>3. To overwrite data from an earlier download attempt, select the Re-import data check box.</td>
</tr>
<tr>
<td></td>
<td>4. Click Download.</td>
</tr>
</tbody>
</table>
The following events happen when the job executes:

- After downloading the data, Cloud Insights updates the billing node data table (sn_cld_intg_aws_cost_usage) with the CIs in the CMDB that correspond to each resource ID. If a CI does not exist, the system generates a placeholder CI. On subsequent discovery, the system reconciles the placeholder CI. (The CI placement process associates downloaded cost and usage data with the appropriate CIs. See Add a class type to the Billing query table for details.)

- Cloud Insights generates a log entry for each stage of the execution. See Monitoring executions of download and reporting jobs for details on the logged information.
Cancel a Billing Download job

A Billing Download job downloads billing data from each payer account for each provider. You can cancel any Billing Download job individually.

Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to **Cloud Insights > AWS Settings > Billing Download Jobs**.
2. Click the execution to cancel (either currently running or scheduled).
3. Cancel the execution:
   - To cancel a currently running execution: On the Billing Download Job form, click **Cancel Execution**.
   - To cancel a scheduled execution:
     1. On the Billing Download Job form, click the execution in the Scheduled Executions related list.
     2. Click **Cancel**.

Add a class type to the Billing query table

The CI placement process is a part of Billing Download job execution. The CI placement process associates downloaded cost and usage data with the appropriate CIs. Specify the details of a new CI or CI type to enable the process to correctly assign cost and usage data.

Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

The Billing Download jobs store downloaded billing data in the billing node table. The Class Type table holds CI type definitions for the cost usage table. Cloud Insights executes the CI placement process to associate the billing and usage data with CIs in the CMDB. The CI placement process queries the billing node table using the CI type definitions that reside in the CI placement type table.

The AWS CI placement type table (sn_cld_intg_aws_ci_placement_type) that is included in the Cloud Insights base system inherits from the Core CI placement type table, shown here:
The following procedure describes how to add a new class type definition to an existing CI placement type table. The step examples use the AWS table (sn_cld_intg_aws_ci_placement_type_list).

1. Go to the CI placement type table and click **New**.
2. Fill in the form.
### CI Placement Type Information Form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class type</td>
<td>The class type table that this type of CI is stored in. New CIs of this type are added to the table.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the CI type.</td>
</tr>
<tr>
<td>Query table</td>
<td>The billing node table that the CI placement process queries to find instances of the CI type.</td>
</tr>
<tr>
<td>Query params</td>
<td>The parameter values in the billing node table that uniquely define the CI type.</td>
</tr>
</tbody>
</table>

3. Add a new CI placement type table.
   a) Inherit from (sn_cld_intg_core_ci_placement_type) and call it (for example):
      [sn_cld_intg_<provider-name>_ci_placement_type]
   b) Add or modify class type definitions as needed.

### Schedule and manage the jobs that download price sheets

A Price Sheet Download job downloads and stores price sheet data. Rightsizing processes use price sheet data when generating rightsizing recommendations.

- For each provider, run Discovery on each service account.
• Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

On the Home page, when you click the **Set up Download Schedule** button in the Configure Price Sheets section, the Price Sheets Download Jobs form opens.

• You create a scheduled Price Sheet Download job for each provider.
• Each execution of a Price Sheet Download job runs multiple execution items. Each execution item imports and stores the price sheet for one region.
• To ensure accurate reporting and recommendations, make sure that Discovery runs before the scheduled execution.
• When the scheduled time arrives, job execution happens in multiple stages (for example, connect to the provider, download the data, perform post-import sort, and so on). The system logs status and results on the Job Executions dashboard for each stage.

### Tables that store price sheet data

<table>
<thead>
<tr>
<th>Table type</th>
<th>Tables</th>
</tr>
</thead>
</table>
| Cloud integrations AWS tables   | • sn_cld_intg_aws_price_sheet_info  
                                | • sn_cld_intg_aws_product_details_ec2  
                                | • sn_cld_intg_aws_product_offering_ec2 |
| Cloud integrations core tables  | • sn_cld_intg_core_ps_schedule  
                                | • sn_cld_intg_core_ps_download_request  
                                | • sn_cld_intg_core_ps_download_request_items |

1. During initial installation, on the Home page, when you click the **Set up Download Schedule** button in the Configure Price Sheets section, the Price Sheets Download Jobs form opens. After initial setup, you navigate to **Cloud Insights > AWS Settings > Price Sheet Download Jobs**.

2. Click **New** and then fill in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Meaningful name for the Price Sheet Download job.</td>
</tr>
<tr>
<td>Provider</td>
<td>The cloud provider.</td>
</tr>
<tr>
<td>Last successful execution</td>
<td>Read-only field. The timestamp of the most recent execution of the job.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for activating the job. Only active jobs are executed.</td>
</tr>
<tr>
<td>Run</td>
<td>Frequency to execute the job. Default: Daily</td>
</tr>
<tr>
<td>Time</td>
<td>Time of day to execute the job.</td>
</tr>
<tr>
<td>Current execution status</td>
<td>Read-only field. The status of the execution that is currently running.</td>
</tr>
</tbody>
</table>

**Note:** Depending on the value that you select, additional fields appear. For example, if you select a **Run** value of **Monthly**, a **Day** field appears. A value of 3 would mean the third day of the month.
### 3. You can execute the job immediately or save the scheduled job:

<table>
<thead>
<tr>
<th>Choice</th>
<th>Resulting actions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Click Submit.</strong></td>
<td>Create the schedule for the Price Sheet Download job.</td>
</tr>
<tr>
<td><strong>Click Execute Now.</strong></td>
<td>Execute the Price Sheet Download job immediately.</td>
</tr>
</tbody>
</table>

The Price Sheet Download Job dialog box opens.

1. In the **Start month** field, enter the first month of data to download.
2. In the **End month** field, enter the last month of data to download.
3. To overwrite data from an earlier download attempt, select the **Re-import data** check box.
4. Click **Download**.

During execution, Cloud Insights downloads and stores the data. The app logs status and results on the Job Executions dashboard for each execution item. If there is no new data, the execution is marked Skipped. If the download process is stuck, the execution is marked as Canceled. See [Monitoring executions of download and reporting jobs](#) for details on the logged information.

### Cancel a Price Sheet Download job

A Price Sheet Download job downloads price sheet data from each provider. You can cancel any Price Sheet Download job individually.

- For each provider, run Discovery on each service account.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

Cloud Insights downloads price sheet data from each provider one region at a time. If you cancel a running price sheet download job, the current region finishes downloading and the system cancels download for the remaining regions. If you cancel a scheduled job execution, then all regions are marked as canceled.

1. **Navigate to Cloud Insights > AWS Settings > Price Sheet Download Jobs.**
2. **Click the job execution to cancel** (either currently running or scheduled).
3. **Cancel the execution:**
   - To cancel a currently running execution: On the Price Sheet Download Job form, click **Cancel Execution**.
   - To cancel a scheduled execution:
     1. On the Price Sheet Download Job form, click the execution in the Scheduled Executions related list.
     2. **Click Cancel**.

---

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Exclude a resource from Cloud Insights reports

To ensure that cost data for a particular resource does not appear in a report, you exclude the resource by adding it to the Excluded Resource list.

- A cloud account with a service account and associated datacenters is required.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

- Excluding a resource from a report means only that the resource will not appear in the report. This setting does not affect analysis of data for the resource.
- At any time, you can unexclude a resource (remove the resource from the Excluded Resources list).
- Production resources are examples of resources that you might exclude. Because production resources must always be active, you might, for example, want to exclude production resources from Business Hours reports.

Exclude a resource either from a single report type or from all reports.

- Exclude a resource from a single report type: On any Resources or Recommendation tab, select the resource and click Exclude. The resource is excluded from the current report and is added to the Excluded Resources list for the report.
- Exclude a resource from all reports (add a resource to the global Excluded Resources list):
  1. Navigate to Cloud Insights > Global Settings > Global Exclusions.
  2. On the Excluded Resources page, click New and fill in the form. Click Save. The resource will then be excluded from all Cloud Insights reports.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>The resource to exclude. Only VM resources appear in the list.</td>
</tr>
<tr>
<td>Description</td>
<td>The reason for excluding the resource.</td>
</tr>
</tbody>
</table>

Remove a resource from the Excluded Resource list

At any time, you can unexclude a resource (remove the resource from the Excluded Resources list).

- A cloud account with a service account and associated datacenters is required.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

When you remove a resource from the Excluded Resources list, the resource appears in the appropriate reports.

Remove a resource from the Excluded Resources list.

- To remove a resource from the Excluded Resources list for a single report type: On the Excluded Resources tab for the report, select the resource and click Remove from Excluded Resources list.
- To remove a resource from the global Excluded Resources list:
  1. Navigate to Cloud Insights > Global Settings > Global Exclusions.
  2. Select the resources to remove.
3. In the Actions on selected rows list, select **Delete** and then confirm deletion.

**Configuring Cloud Insights system properties**

Some properties are available on a system properties form, but some lesser-used properties are available only from the System Property (sys_properties) table.

**System properties for Billing Download jobs**

**sn_cld_intg_core.ci_placement_threads_number**

Number of threads that are used by CI placement flow to create CIs in the billing process.
- **Type:** integer
- **Default value:** 4
- **Location:** System properties

**sn_cld_intg_aws.max_unprocessed_records_for_ci_placement**

Maximum number of records to read in one batch from the AWS Cost And Usage Bill Data table for CI Placement.
- **Type:** integer
- **Default value:** 300000
- **Location:** System properties

**sn_cld_intg_aws.min_unprocessed_records_for_ci_placement**

Minimum number of records to read in one batch from the AWS Cost And Usage Bill Data table for CI Placement.
- **Type:** integer
- **Default value:** 100000
- **Location:** System properties

**System properties for Spend analysis**

**sn_cld_spend_core.metricbase_transform_limit**

Maximum number of Billing records to use in one Metric Base transform.
- **Type:** integer
- **Default value:** 100000
- **Location:** System properties

**sn_cld_spend_core.spend_report_flow_launcher_chunk_size**

Number of non-empty CI chunks to send to the spend flow launcher.
- **Type:** integer
- **Default value:** 40
- **Location:** System properties
**sn_cld_spend_core.spend_report_per_chunk_workload_size**
Number of workloads (CI ranges and metadata) for each chunk in the spend flow launcher.
- Type: integer
- Default value: 1000
- Location: System properties

**sn_cld_spend_aws.num_months_forecast**
Number of future months after the current month for which the AWS Forecast spend is retrieved during every execution of the Spend job.
- Type: integer
- Default value: 2
- Location: System properties

**System properties for Business Hours**

Note: The collection of data for a single download of billing or price sheet data is called a *workload*. Data in a workload is held in sets of data called *chunks*. You can specify the size of each chunk and the number of chunks in a workload.

**sn_clin_core.bh_report.batch_size**
The number of resources to include in a Business Hours report batch.
- Type: integer
- Default value: 5000
- Location: System properties

**sn_clin_core.bh_recom_max_chunk_size**
Number of records used to create workload chunks of rightsizing recommendations. If you set a value greater than `glide.db.max_view_records`, then the `glide.db.max_view_records` value is used instead. A smaller number means more chunks and a larger number means fewer chunks.

Records considered in one execution of generating recommendations = (number of chunks) * (data records/chunk)
- Type: integer
- Default value: 10000
- Location: System properties

**sn_clin_core.bh_recom_num_chunks_per_workload**
Number of chunks per workload. This value times the value of `sn_clin_core.bh_recom_max_chunk_size` is the total number of records per workload when generating rightsizing recommendations.

Records considered in one execution of generating recommendations = (number of chunks) * (data records/chunk)
- Type: integer
- Default value: 10
Cloud Insights admin guide

The Cloud Insights application, available in the ServiceNow Store, helps you to analyze the full range of costs associated with cloud assets so you can identify and take action on opportunities to save money and optimize operations.

How Cloud Insights works

Cloud Insights continuously monitors your cloud infrastructure to analyze costs so you can identify and take action on resources to save money and optimize operations.

Benefits for Executives

To help ensure that your cloud usage is in line with your organization's budget, vision and goals, Cloud Insights provides the following benefits:

- Accurate, current, and easy-to-digest information.
- The ability to tie cloud use (what is used and when, who has access and for how long) into measurable outcomes, which enables you to predict the need for investment and to meet and exceed customer demand with timely releases.

Benefits for Cloud Administrators

You need to set up and monitor cloud usage, to track metrics such as changes in integration or the number of code or infrastructure changes, to review forecasting and weekly trends to determine better provisioning strategies, to share usage reports with management and execs, and to shut down or rights size resources when appropriate. Cloud Insights enables you to accomplish the following tasks:

- Set up data collection and usage policies.
- Monitor usage and shut down or rights size resources as needed.
- Design custom reports that display clear and actionable information.
- Confidently share reports with management.
- View weekly trends with forecasting tools.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Benefits for Cloud Managers

You need to define project and portfolio goals, understand how and why resources are being provisioned, monitor and act on burn rates, forecast your cloud spend and then share cost and forecast info with your team and with execs. Cloud Insights lets you accomplish those tasks:

- View dashboards with burn rates and insight into events and trends on costs and usage.
- Design dynamic reports for execs and upper-level managers.
- Proactively identify potential issues and then take action to avoid them.

Questions that you need answers to

How much am I spending in the public cloud? Am I paying for resources that are over-provisioned or underused? Am I paying for resources that have no known function (or are unused)?

- Cloud Spend Analytics reports help you to identify opportunities to reduce cloud costs. See the Cloud spend: Last 30 Day Spend reports, Cloud spend: Monthly Spend reports, and Cloud spend: Forecast reports.
- The Rightsizing feature analyzes resource usage to recommend better sizes for resources that are wasting money by being over-provisioned or underused. A confidence rating and predicted savings support each recommendation to rightsize a resource. See Rightsizing: Implementing cost-effective sizing for resources.

Do all my cloud assets follow my policies? Which assets can I not track accurately because they are not properly tagged with an assignment group? Am I paying for resources that are not "owned" by a cost center or department?

To ensure that the information on usage and spending is accurate, Unassigned Resources policies identify resources that do not belong to an assignment group. For such resources, the policy can auto-assign an assignment group to a resource, require a member of the assignment group to approve the assignment, or add the resource to a report and take no further action. See Identifying unassigned resources to ensure accurate reporting.

Which assets are operating outside the hours that they should be operating?

A Business Hours job applies Business Hours policies to identify resources that are running when they should be powered off, reports them, and can start and stop them on the schedule that you specify. Running only during specified business hours can significantly reduce your cloud spend. See Business Hours: Ensuring resources run only when required.

Service category names in Cloud Insights and Amazon AWS Cloud

Use this information to determine the name of the AWS service category (resource type) that corresponds to a service category name that appears in Cloud Insights.

<table>
<thead>
<tr>
<th>Cloud Insights Service Category</th>
<th>Services in Amazon AWS Cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analytics</td>
<td>Amazon Elasticsearch Service</td>
</tr>
<tr>
<td>Analytics</td>
<td>Amazon EMR</td>
</tr>
<tr>
<td>Analytics</td>
<td>AWS Data Pipeline</td>
</tr>
<tr>
<td>Analytics</td>
<td>Amazon CloudSearch</td>
</tr>
<tr>
<td>Cloud Insights Service Category</td>
<td>Services in Amazon AWS Cloud</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Analytics</td>
<td>AWS Glue</td>
</tr>
<tr>
<td>Analytics</td>
<td>Amazon Kinesis</td>
</tr>
<tr>
<td>Analytics</td>
<td>Amazon QuickSight</td>
</tr>
<tr>
<td>Analytics</td>
<td>Amazon Athena</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon MQ</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Elemental MediaStore</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon API Gateway</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon Connect</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon Elastic Transcoder</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Step Functions</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon Sumerian</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon Kinesis Video Streams</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Application Discovery Service</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon Simple Email Service</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS (Unknown Product)</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon SageMaker</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon GameLift</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS DataSync</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Elemental MediaConvert</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Elemental MediaPackage</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Elemental MediaTailor</td>
</tr>
<tr>
<td>Application Services</td>
<td>AWS Elemental MediaLive</td>
</tr>
<tr>
<td>Application Services</td>
<td>Amazon Simple Queue Service</td>
</tr>
<tr>
<td>Business Productivity</td>
<td>Alexa for business</td>
</tr>
<tr>
<td>Business Productivity</td>
<td>Amazon Chime</td>
</tr>
<tr>
<td>Business Productivity</td>
<td>Amazon WorkDocs</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS Lambda</td>
</tr>
<tr>
<td>Compute</td>
<td>Amazon Lightsail</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS Lambda</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS Fargate</td>
</tr>
<tr>
<td>Compute</td>
<td>Amazon Elastic Container Registry</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS Elastic Compute Cloud</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS Elastic Beanstalk</td>
</tr>
<tr>
<td>Compute</td>
<td>Amazon EC2 Auto Scaling</td>
</tr>
<tr>
<td>Compute</td>
<td>AWS CloudFormation</td>
</tr>
<tr>
<td>Compute</td>
<td>Amazon Elastic Compute Cloud</td>
</tr>
<tr>
<td>Compute</td>
<td>Amazon EC2 Container Service</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
<table>
<thead>
<tr>
<th>Cloud Insights Service Category</th>
<th>Services in Amazon AWS Cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compute</td>
<td>AWS Batch</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon Aurora</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon ElastiCache</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon SimpleDB</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon Redshift</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon Relational Database</td>
</tr>
<tr>
<td>Database</td>
<td>AWS Relational Database</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon DocumentDB</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon Neptune</td>
</tr>
<tr>
<td>Database</td>
<td>Amazon DynamoDB</td>
</tr>
<tr>
<td>Database</td>
<td>AWS Database Migration Service</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS Cloud9</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS CodeCommit</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS CodePipeline</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS X-Ray</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS CodeBuild</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS CodeDeploy</td>
</tr>
<tr>
<td>Developer Tools</td>
<td>AWS CodeStar</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS IoT Core</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS IoT Events</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS IoT Device Management</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS IoT 1-Click</td>
</tr>
<tr>
<td>IOT</td>
<td>Amazon FreeRTOS</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS Greengrass</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS IoT Analytics</td>
</tr>
<tr>
<td>IOT</td>
<td>AWS IoT</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>AWS Elastic Load Balancing</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>Elastic Load Balancing</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Lex</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Polly</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Machine Learning</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Translate</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Rekognition</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Transcribe</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Amazon Comprehend</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AWS OpsWorks</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AWS Auto Scaling</td>
</tr>
<tr>
<td>Cloud Insights Service Category</td>
<td>Services in Amazon AWS Cloud</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AWS Service Catalog</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AWS Systems Manager</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AWS Cost Explorer</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AWS Trusted Advisor</td>
</tr>
<tr>
<td>Management Tools</td>
<td>AmazonCloudWatch</td>
</tr>
<tr>
<td>Mobile</td>
<td>Amazon Pinpoint</td>
</tr>
<tr>
<td>Mobile</td>
<td>AWS AppSync</td>
</tr>
<tr>
<td>Mobile</td>
<td>AWS Device Farm</td>
</tr>
<tr>
<td>Mobile</td>
<td>AWS Mobile Hub</td>
</tr>
<tr>
<td>Network</td>
<td>Amazon Virtual Private Cloud</td>
</tr>
<tr>
<td>Network</td>
<td>AWS Cloud Map</td>
</tr>
<tr>
<td>Network</td>
<td>AWS Direct Connect</td>
</tr>
<tr>
<td>Network</td>
<td>Amazon Route 53</td>
</tr>
<tr>
<td>Network</td>
<td>Amazon CloudFront</td>
</tr>
<tr>
<td>Network</td>
<td>AWS Virtual Private Cloud</td>
</tr>
<tr>
<td>Network</td>
<td>AWS Route 53</td>
</tr>
<tr>
<td>Productivity</td>
<td>Amazon Workspaces</td>
</tr>
<tr>
<td>Productivity</td>
<td>Amazon Simple Workflow Service</td>
</tr>
<tr>
<td>Productivity</td>
<td>Amazon WorkMail</td>
</tr>
<tr>
<td>Security</td>
<td>Amazon GuardDuty</td>
</tr>
<tr>
<td>Security</td>
<td>AWS Directory Service</td>
</tr>
<tr>
<td>Security</td>
<td>AWS Secrets Manager</td>
</tr>
<tr>
<td>Security</td>
<td>Amazon Inspector</td>
</tr>
<tr>
<td>Security</td>
<td>Amazon Cognito</td>
</tr>
<tr>
<td>Security</td>
<td>AWS Key Management Service</td>
</tr>
<tr>
<td>Security</td>
<td>Amazon Cloud Directory</td>
</tr>
<tr>
<td>Security</td>
<td>AWS Shield</td>
</tr>
<tr>
<td>Security</td>
<td>AWS Identity and Access Management</td>
</tr>
<tr>
<td>Security</td>
<td>AWS WAF</td>
</tr>
<tr>
<td>Security</td>
<td>Amazon Macie</td>
</tr>
<tr>
<td>Security</td>
<td>AWS CloudHSM</td>
</tr>
<tr>
<td>Security</td>
<td>AWS Certificate Manager</td>
</tr>
<tr>
<td>Storage</td>
<td>AWS Simple Storage Service</td>
</tr>
<tr>
<td>Storage</td>
<td>AWS Elastic Block Storage</td>
</tr>
<tr>
<td>Storage</td>
<td>AWS Snowball</td>
</tr>
<tr>
<td>Storage</td>
<td>Amazon Elastic File System</td>
</tr>
<tr>
<td>Storage</td>
<td>Amazon Simple Storage Service</td>
</tr>
</tbody>
</table>
Cloud Insights home page

After you have configured the Cloud Insights app, the home page displays a scannable overview of areas where you might save money and improve utilization of resources.

Cloud Insights home page

On the home page (Cloud Insights > Home), you can quickly scan costs associated with various aspects of your cloud infrastructure. Just click one of the View buttons to access reports and view job execution status. Cloud Insights reports are scannable dashboards that present trends and forecasts on cloud spending and help you to understand where and how you currently spend on cloud resources.
Cloud Insights

Cloud Insights provides visibility, management, and governance for your cloud infrastructure and your business.

Cloud Spend Analytics
Spend across all providers

$8,771
Last 30 days

Unassigned Resources
Gain visibility on unassigned resources

1056
Unassigned resources
97% of total

1 View Analytics
2 View Details

Rightsizing
Optimize spend and performance

$1,647
Potential 30-day savings
19% of total spend

Business Hours
Set On/Off schedules for resources

$4
Potential 30-day savings
0% of total spend

3 View Details
4 View Details
1. Cloud Spend Analytics reports:
   
   - **Cloud spend: Last 30 Day Spend reports**: The Last 30-Day Spend tab displays a chart of spending for the most recent 30-day period grouped by service category (resource type) or over the time period that you specify. You can filter the data that appears in the chart.
   
   - **Cloud spend: Monthly Spend reports**: The Monthly Spend tab displays an overview of cloud costs. The tab displays a chart of your cloud spend over a date range that you specify. You can filter the data that appears in the chart and group the data by category like region or service category (resource type).
   
   - **Cloud spend: Forecast reports**: The Forecast tab displays forecasted cloud costs for the current month and for upcoming months.

2. **Unassigned Resources overview**: Unassigned Resources policies help you to identify the resources that are not associated with an assignment group and to assign them appropriately. When a resource is assigned to the correct group, the resource can be appropriately governed even as it goes through stages such as patching, upgrading, and reconfiguring. The Unassigned Resources Overview tab highlights the number of resources that do not have an assignment group and the status of your Unassigned Resources policies. The Cloud Discovery process triggers Unassigned Resources analysis and updates recommendations in reports.

3. **Rightsizing overview**: The Rightsizing Overview tab highlights the resources that waste the largest portion of your cloud spend. It also includes reports that show how many resources are scheduled for rightsizing and how many additional resources might benefit from rightsizing.

4. **Business Hours overview**: The Business Hours Overview tab identifies the top five opportunities to improve business-hours usage and costs and the number of resources in each state of adjustment of business-hours.

5. **Job Executions dashboard**: The Job Executions dashboard displays status information on each type of job that Cloud Insights executes. The various jobs download, store, and analyze cost and usage data and generate reports that present actionable recommendations.

---

**Cloud spend: Last 30 Day Spend reports**

The Last 30-Day Spend tab displays a chart of spending for the most recent 30-day period grouped by service category (resource type) or over the time period that you specify. You can filter the data that appears in the chart.

**Viewing the Last 30 Day Spend reports**

Use one of the following methods to view the reports:

- Click Analytics > Cloud Spend and then click the Last 30 Day Spend tab.
- Click Home. In the Cloud Spend Analytics section, click View Details. The Cloud Spend page opens in a new tab. Click the Last 30-Day Spend tab.

**Working with the data**

- All data is auto-updated so there are no settings to configure.
- Click anywhere in a report or chart to view supporting data.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the Service Account filter. The Master Service Account filter limits the data to the AWS payer account that you
specify. You can narrow or widen the range of dates for the data in the chart using the Date Range filter.

- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
- To save a chart as an image, click the menu icon
  
  ![Menu Icon](image)

and select the image type.
Last 30 Day Spend tab

- The Last 30 Day Spend chart shows the full spend for the preceding 30 days, color-coded by service category (resource type). Color codes appear under the Spend by Service Category chart.
• The Spend by Service Category chart shows the spend for each service category for the preceding 30 days. You can use the **Date Range** filter to specify the time period to present on the chart.

**Note:** Because Cloud Insights supports multiple providers, it uses general terms for many service categories. For information about which terms apply, see Service category names in Cloud Insights and Amazon AWS Cloud.

Cloud spend: Monthly Spend reports

The **Monthly Spend** tab displays an overview of cloud costs. The tab displays a chart of your cloud spend over a date range that you specify. You can filter the data that appears in the chart and group the data by category like region or service category (resource type).

**Viewing the Monthly Spend reports**

Use one of the following methods to view the reports:

- Click **Analytics > Cloud Spend** and then click the **Monthly Spend** tab.
- Click **Home**. In the Cloud Spend Analytics section, click **View Details**. The Cloud Spend page opens in a new tab. Click the **Monthly Spend** tab.

**Working with the Monthly Spend data**

- All data is auto-updated so there are no settings to configure.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the **Service Account** filter. The Master Service Account filter limits the data to the AWS payer account that you specify. You can narrow or widen the range of dates for the data in the chart using the Date Range filter.
- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
- Click anywhere in a report or chart to view supporting data.
- To save a chart as an image, click the menu icon and select the image type.
### Monthly Spend tab

#### Current Month Spend
- **$4,242.00**

#### Last Month Spend
- **$6,803.00**
- The Last Month Spend value is the value from Amazon Web Services of your costs for the entire previous month.
- The Month-by-Month Spend chart shows the full spend up to today for the current month, color-coded by the setting that you specify for the Group by filter. Descriptions of the color codes appear under the chart. Specify the time period of data to display using the Date Range filter.
- Click the menu icon

```
(   )
```
to save a chart as an image.

Cloud spend: Forecast reports
The Forecast tab displays forecasted cloud costs for the current month and for upcoming months.

Viewing the reports

Use one of the following methods to view the reports:
- Navigate to Cloud Insights > Analytics > Cloud Spend and then click the Forecast tab.
- Navigate to Cloud Insights > Home. In the Cloud Spend Analytics section, click View Details. The Cloud Spend page opens in a new tab. Click the Forecast tab.

Working with the data

- All data is auto-updated so there are no settings to configure.
- The Master Service Account filter limits the data to the AWS payer account that you specify.
- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
- Click anywhere in a report or chart to view supporting data.
Cloud Spend Forecast tab

Current Month Forecast

$8,372.00

Next Month Forecast

$7,142.00

Month-by-Month Forecast

- Oct/2019
- Nov/2019
- Dec/2019
• The Next Month Forecast value is your expected cost for the entire following month. Amazon Web Services calculates the forecast based on past costs.
• The Month-by-Month Forecast chart shows the projected costs for the next three months.
• Click the menu icon

( )
to save a chart as an image.

Spend optimization: Overall spend reports

The Cloud Insights Optimization reports present trends and forecasts on cloud spending and help you to understand where and how you currently spend on cloud resources. The Overall tab identifies and ranks cloud spend by owners and groups.

Working with the data

• All data is auto-updated so there are no settings to configure.
• Click anywhere in a report or chart to view supporting data.
• All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
• By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the Service Account filter. You can narrow or widen the range of dates for the data in the chart using the Date Range filter.
• To save a chart as an image, click the menu icon

( )
and select the image type.

Spend Optimization: Overall tab

Navigate to Cloud Insights. Click Analytics > Spend Optimization and then click the Overall tab. The Overall tab identifies and ranks cloud spend by owners and groups.
**Total Cloud Spend**
Spend for the last 30 days.

**Top Cloud Spend by Assignment Group**
Ordered list of assignment groups that spent the most over the last 30 days.

**Note:** Cloud Insights uses the Assignment Group value of a resource to show trends in spending. The Unassigned Resources feature identifies resources that are not associated with an assignment group so you can assign them. See [Identifying unassigned resources to ensure accurate reporting](#).

**Top Cloud Spend by Owner**
List of resource owners that spent the most over the last 30 days.

**Spend optimization: Business Hours reports**
The Business Hours reports help you to identify resources that waste money by being powered on for a significant time when they should be powered off.

**Viewing the reports**
Use one of the following methods to view the reports:

- Click Analytics > Spend Optimization and then click the Business Hours tab.
- Click Cost Optimization > Business Hours. On any tab on the Business Hours page, click Business Hours Analytics. The tab opens in a new browser tab.
Working with the data

- All data is auto-updated so there are no settings to configure.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the Service Account filter. The Master Service Account filter limits the data to the AWS payer account that you specify. You can narrow or widen the range of dates for the data in the chart using the Date Range filter.
- Click anywhere in a report or chart to view supporting data.
- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
- To save a chart as an image, click the menu icon and select the image type.

Business Hours tab

In this example, the filter settings limit the reports to show data only for the service account named Insight aws and only for resources that match the AutoApply policy.
$6,178.77  $10.00  $22.00  10
Total cloud spend. The Policy filter does not apply for this chart.

**Total Business Hours Spend**
Total spend for resources that are on during business hours (as determined by Business Hours policies).

**Total Non-Business Hours Spend**
Total spend for resources that are on during non-business hours (as determined by Business Hours policies).

**Total Resources**
Number of resources in your cloud infrastructure.

**Business vs. Non-Business Hours Spend**
Timeline of spend for resources. The report groups spend during business hours and during non-business hours.

**Top Predicted Savings by Owner**
Ordered by resource owner: The biggest savings that could result if all recommended business hours changes are implemented.

**Top Predicted Savings by Assignment Group**
Ordered by assignment group for resources: The biggest savings that could result if all recommended business hours changes are implemented.

**Cost optimization: Rightsizing reports**
Reports on the Rightsizing tab help you to identify the users, user groups, or locations that are wasting money by running over-provisioned or underused resources. Other reports on the tab present your expected savings if you were to take action to rightsize the identified resources.

**Viewing the reports**
Use one of the following methods to view the reports:

- Click Analytics > Spend Optimization and then click the Rightsizing tab.
- Click Cost Optimization > Rightsizing. On any tab on the Rightsizing page, click Rightsizing Analytics. The tab opens in a new browser tab.

**Working with the data**
- All data is auto-updated so there are no settings to configure.
- Click anywhere in a report or chart to view supporting data.
- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the Service Account filter. You can narrow or widen the range of dates for the data in the chart using the Date Range filter.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
- To save a chart as an image, click the menu icon and select the image type.

**Rightsizing tab**

![Rightsizing tab](image)

**Total Cloud Spend**

Total cloud spend. The **Job** filter does not apply to this chart.
**Total Predicted Savings**

Total savings to expect if all rightsizing recommendations are applied.

**Total Recommended Resources**

Number of resources that have a rightsizing recommendation.

**Rightsizing Monthly Savings**

Each of the possible states of a rightsizing recommendation appears as a bar on the chart:

- **New**: Potential savings for resources that the most recent analysis has recommended for rightsizing.
- **In-progress**: Potential savings for resources where the Rightsizing change request has not yet been approved.
- **Rejected**: Potential savings for resources where the Rightsizing change request was rejected.
- **Failed**: Potential savings for resources where the Rightsizing change request failed.

**Rightsizing Resources**

Each of the possible states of a rightsizing recommendation appears as a bar on the chart:

- **New**: Potential savings for resources that the most recent analysis has recommended for rightsizing.
- **In-progress**: Number of resources where the Rightsizing change request has not yet been approved.
- **Rejected**: Number of resources where the Rightsizing change request was rejected.
- **Failed**: Number of resources where the attempt to rightsize the resource failed.

**Top Predicted Savings by Owner**

Each bar represents the potential savings for a resource owner.

**Top Predicted Savings by Assignment Group**

Each bar represents the potential savings for an assignment group.

**Monitoring executions of download and reporting jobs**

The Job Executions dashboard displays status information on each type of job that Cloud Insights executes. The various jobs download, store, and analyze cost and usage data and generate reports that present actionable recommendations.

**Reports on the Job Executions dashboard**

The Job Executions dashboard displays reports on the jobs that Cloud Insights executes.

- **Billing Download jobs** download, organize, and store billing data for each payer account. The system analyzes the data to make recommendations for changes in your cloud operations that can lead to cost savings.
- **A Price Sheet Download job** downloads and stores price sheet data. Rightsizing processes use price sheet data when generating rightsizing recommendations.
- **Spend jobs** analyze cost and usage data to generate analytics reports on your cloud costs and trends.
- **Each update of billing data** triggers the Business Hours and Rightsizing jobs to analyze spend and usage data and to update the actionable recommendations in reports.
- **The Cloud Discovery process** triggers Unassigned Resources analysis and updates recommendations in reports.
To specify the time period that the reports should cover, use the Date Range filter.
Click any report to view status and log data for executions.

Accessing the Job Executions dashboard

Use one of the following methods to view the reports:

- Navigate to Cloud Insights > Monitoring > Job Executions.
- Navigate to Cloud Insights > Home and then click the Job Executions button.
<table>
<thead>
<tr>
<th>Date Range</th>
<th>Billing Downloads</th>
<th>Price Sheet Downloads</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>In Progress</th>
<th>Failed</th>
<th>In Progress</th>
<th>Failed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Billing Downloads</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Price Sheet Downloads</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Canceled</th>
<th>Completed</th>
<th>Canceled</th>
<th>Completed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Billing Downloads</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Price Sheet Downloads</td>
<td>1</td>
<td>8</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Spend</th>
<th>Rightsizing</th>
</tr>
</thead>
<tbody>
<tr>
<td>In Progress</td>
<td>Failed</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Canceled</td>
<td>Completed</td>
</tr>
<tr>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Business Hours</th>
<th>Unassigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>In Progress</td>
<td>Failed</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Reports on the Job Executions dashboard

<table>
<thead>
<tr>
<th>Job execution type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Billing Downloads</td>
<td>A Billing Download job is a series of processes that download and store billing data. Each successful execution updates the billing data tables and CIs in the CMDB and sets off Spend, Rightsizing, and Business Hours executions.</td>
</tr>
<tr>
<td>Price Sheet Downloads</td>
<td>A Price Sheet Download job downloads and stores price sheet data.</td>
</tr>
<tr>
<td>Spend</td>
<td>A Spend execution is a series of processes that analyze spend and usage data and then generate Spend reports. Each update of billing data triggers the Business Hours and Rightsizing jobs.</td>
</tr>
<tr>
<td>Rightsizing</td>
<td>A Rightsizing execution is a series of processes that analyze spend and usage data, apply pricing discounts, and generate rightsizing recommendations and update reports. Each update of billing data triggers the Business Hours and Rightsizing jobs. For information on specifying discounts that the provider applies to your organization, see Specify rate discounts to enable accurate pricing for rightsizing recommendations.</td>
</tr>
<tr>
<td>Business Hours</td>
<td>A Business Hours execution is a series of processes that apply policies to spend and usage data and generate Business Hours recommendations and update reports. Each update of billing data triggers the Business Hours and Rightsizing jobs.</td>
</tr>
<tr>
<td>Unassigned</td>
<td>An Unassigned execution is a series of processes that apply policies to usage data, generate Unassigned Resources recommendations and update reports. The Cloud Discovery process triggers Unassigned Resources analysis and updates recommendations in reports.</td>
</tr>
</tbody>
</table>

**Job execution logs**

When you click any report on the dashboard, the list of execution records with the selected job execution status appears.

- In Progress: The job is executing (running).
- Failed: At least one of the processes that the job performs failed to run to completion.
- Canceled: A user canceled the job before it could run to completion.
- Completed: The job ran to completion successfully.

**Billing Download jobs**

The following example shows a Billing Download Executions table for a failed execution. The execution records can help you to determine why a particular execution failed. Each execution has an associated progress bar.
The execution of a billing job proceeds in several stages. Each stage performs one or more actions on the data (import data, sort the data, update the CMDB, store the data, and so on). For example, the Post-Import of Billing Data stage reconciles the resource identifiers from the billing data with the corresponding CIs in the CMDB. Each execution record in the list identifies the stage that is currently running or the last stage that ran. Click an execution Start time to view the stages that make up the execution. In the following example, the execution failed in the Importing Billing Data stage.
Click a stage to view its details. Click an extension to view the execution details for the extension.
The Extension Executions (Download Cloud Integrations AWS Cost Usage Report and CI Placement Extension) show the extensions that ran as part of the stage. Each stage has logs and each log has a level.
### Execution Stage Logs (31)

<table>
<thead>
<tr>
<th>Stage</th>
<th>Start time</th>
<th>End time</th>
</tr>
</thead>
</table>

**Message**

Error during extension execution - [Flow for extension: "CI Placement Trigger Extension", is in state "CANCELLED". Check the execution "https://crmdb07.service-now.com/$flow-designer.do#/operations/context/d7071d438ea88410c63588770e36399c" for more information]

**Extension Executions (2)**

<table>
<thead>
<tr>
<th>Timestamp</th>
<th>Level</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>2019-10-21 14:34:39</td>
<td>Information</td>
<td>450,000 records processed.</td>
</tr>
</tbody>
</table>
| 2019-10-21 14:34:39 | Information | **** Summary ****  
Number of records processed: 450,000  
Status: error  
Metrics Processed: ("u_cost":4400.6304335840371)  
Time taken: 7.36 minutes |
| 2019-10-21 14:34:39 | Information | 450,000 records processed.                                                                             |
Price Sheet Download Executions logs

Click an execution in the list to view the execution items for the execution.

**Note:** The End time field is populated even if the execution hasn't reached an ending state. In this case, the value represents the last time the execution stage record was updated with duration, progress, and percent complete information.

<table>
<thead>
<tr>
<th>Number</th>
<th>Cloud Provider</th>
<th>Execution Type</th>
<th>Status</th>
<th>Start Time</th>
<th>End Time</th>
<th>Duration</th>
<th>Progress</th>
<th>Parent Execution</th>
</tr>
</thead>
<tbody>
<tr>
<td>000001812</td>
<td>AWS</td>
<td>AWS price sheet download</td>
<td>Failed</td>
<td>2019-09-27 10:12:17</td>
<td>2019-09-27 10:12:17</td>
<td>0 Seconds</td>
<td>100% (empty)</td>
<td></td>
</tr>
<tr>
<td>000001975</td>
<td>AWS</td>
<td>AWS price sheet download</td>
<td>Failed</td>
<td>2019-09-28 00:20:20</td>
<td>2019-09-28 00:29:47</td>
<td>5 Hours 49 Minutes</td>
<td>4.76% (empty)</td>
<td></td>
</tr>
<tr>
<td>000002069</td>
<td>AWS</td>
<td>AWS price sheet download</td>
<td>Failed</td>
<td>2019-09-29 00:10:21</td>
<td>2019-09-29 00:10:21</td>
<td>0 Seconds</td>
<td>100% (empty)</td>
<td></td>
</tr>
</tbody>
</table>

The execution of a Price Sheet Download Execution job proceeds as multiple execution items. Each execution item imports and stores the price sheet for one region.
### ServiceNow Execution Details

<table>
<thead>
<tr>
<th>Execution ID</th>
<th>PDE0001019</th>
</tr>
</thead>
<tbody>
<tr>
<td>Publication Date</td>
<td></td>
</tr>
<tr>
<td>Start Time</td>
<td>2019-10-15 08:06:53</td>
</tr>
<tr>
<td>Status</td>
<td>Canceled</td>
</tr>
</tbody>
</table>

#### Execution Items

<table>
<thead>
<tr>
<th>Region</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>me-south-1</td>
<td>Completed</td>
</tr>
<tr>
<td>eu-central-1</td>
<td>Completed</td>
</tr>
<tr>
<td>sa-east-1</td>
<td>Completed</td>
</tr>
</tbody>
</table>
Customizing the Job Executions dashboard

Because the Job Executions dashboard is a standard report, you can change the reports or add or rearrange widgets as needed. Click the + icon to customize the page. If you customize the page, keep in mind that the invisible building block near the top of the page is there to ensure that the rows in the reports line up neatly.

Business Hours: Ensuring resources run only when required

A Business Hours job applies Business Hours policies to identify resources that are running when they should be powered off, reports them, and can start and stop them on the schedule that you specify. Running only during specified business hours can significantly reduce your cloud spend.

How the Business Hours feature works

Each update of billing data triggers the Business Hours and Rightsizing jobs to analyze spend and usage data and to update the actionable recommendations in reports. Business Hours jobs follow this process:
Define a business hours policy

Finance department policy
Applies to:
Resources tagged "finance"
Business hours schedule:
8:00 - 5:00
Policy actions:
Apply business schedule to every matching VM

Admin

Apply the policy
Find all the resources in the CMDB that match the policy conditions

System

Repeat this process when new billing data appears

Perform the actions specified by the policy
Apply the business hours schedule to each matching resource

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
1. For each Business Hours policy, examine the CMDB to identify resources that match the policy criteria. (Policies are described in the following sections.)

2. For each resource that matches a policy, generate a change request to enforce the on/off schedule that is defined in the policy. Alternatively, generate a single change request that applies to all CIs that match the policy.

3. Update Business Hours reports with resources that match a policy and with approved, rejected, and failed changes.

4. When a change request is approved, apply the on/off schedule that is defined in the policy to the associated resources.

5. Repeat this process whenever billing data is updated.
Business Hours policy

Policy name: Finance department

Policy is: Active

Policy type: Auto-apply after approval

Business hours schedule: 8:00 AM to 5:00 PM

Power-on flow: Launch VM Start

Power-off flow: Launch VM Stop

Resource criteria:

Service Accounts
Specify the accounts to apply the policy to.

Providers

AWS

Service accounts
Billing 15970

Resources
Specify the tags that determine which resources match the policy.

This condition must be met

Department is Finance
The Finance department policy example has the following settings:

- The policy is active — it is applied every time billing data is updated.
- The policy type is Auto-apply after approval, which means that, after a qualified user approves the change request, the schedule for resources that match the policy is adjusted.
- The Business Hours (on/off) schedule is "on" from 8:00 AM to 5:00 PM.
- The power-on and power-off flows are specified.
- Only resources that meet the following resource criteria match the policy:
  - The cloud provider is AWS and the service account is Billing 15970.
  - The resource has a tag with the name Department with the value Finance.

After the policy is applied, for each resource that matches the policy (and the change request is approved), the system starts the resource at 8:00AM and stops it at 5:00PM.

**Note:** The actual start and stop times vary slightly due to changes in system demand and the time it takes for the resource to start and stop.

### Business Hours policy types

For each resource that matches the policy criteria, the actions that the policy takes depend on the policy type:

**Auto-apply policy type**

- Generate a recommendation to apply the specified business hours and add the resource to Business Hours reports.
- Generate and auto-approve a change request for the assignment group.
- Apply the Business Hours schedule to the resource.

**Auto-apply after approval**

- Generate a recommendation to apply the specified business hours and add the resource to Business Hours reports.
- Generate a change request for members of the assignment group.
- Add the resource to Business Hours reports.
- Any member of the group with the ITIL role can approve the schedule (through the change request).
- When approved, apply the Business Hours schedule to the resource.

**Report-only**

- Generate a recommendation to apply the specified business hours.
- Add the resource to Business Hours reports.

**Configure Business Hours operations**

You can choose to use or to disable Business Hours features and specify the amount of non-business hour spend that triggers notifications to users or groups. Controlling unnecessary resource use can help reduce costs.

- A cloud account with a service account and associated datacenters is required.
- Role required: Cloud Insights Admin [sn_clin_core.insights_admin]

1. Navigate to **Cloud Insights > Cost Optimization > Business Hours**.
2. On any tab, click **Settings** and then configure the following settings as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable business hours</td>
<td>Option for enabling the Business Hours feature. All active policies are applied when data becomes available. Deselect the check box to disable the Business Hours feature. No policies are applied regardless of the Active setting for a policy.</td>
</tr>
</tbody>
</table>

**Note:** When you disable the Business Hours feature, all schedules that were applied by policies are deactivated and resources return to their original schedules.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notify when unassigned resources exceed (%) of total CIs</td>
<td>The percentage of overall cloud spend for non-business hour usage that triggers email notification of the users or groups that you specify.</td>
</tr>
<tr>
<td>Notify users / Notify groups</td>
<td>The users or groups to notify by email when the non-business hour spend exceeds the specified percentage of overall cloud spend. If you do not specify users or groups, then no notification is sent.</td>
</tr>
</tbody>
</table>
3. Optional: Specify whether to generate a single change request that applies to all CIs that
match a policy (the default setting) or a separate change request for each CI.
   a) Select the Advanced check box.
   b) Select or clear the Single change request check box.

<table>
<thead>
<tr>
<th>Setting of the Single change request check box</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected (default)</td>
<td>The system generates a single change request that applies to all CIs that match the policy. For the Auto-apply after approval policy type, the Maximum CIs per change field appears. Specify the maximum number of resources to associate with a single change request. If the maximum is reached, the system generates a new change request.</td>
</tr>
<tr>
<td>Not selected</td>
<td>The system generates a change request for each CI that matches the policy.</td>
</tr>
</tbody>
</table>

4. Optional: Follow this procedure to create a custom script include.
   The default change request templates that the system uses to generate change requests work well in most applications. Some attributes of the change request template have static/constant values that cannot be changed. Some attributes (for example, risk and work_notes) have dynamic values that you can set using a script include.
   a) Select the Advanced check box.
   b) Select the tab for the policy type to configure.
   c) Specify the script include that sets values in the change template that is used for the change requests for resources that match the policy.
      To change or add values in the template, create a new script include based on one of the default script includes. In the new script include, override the settings in the setValues function. Creating a new script include simplifies upgrades.
      - Auto-apply policies (The Assignment group is derived from the Standard Change template and cannot be overridden): Default: CLINBHStandardChangeRequestUtil
      - Auto-apply after approval policies: Default: CLINBHNormalChangeRequestUtil

5. Click Submit or Update.

Define or update a Business Hours policy
A Business Hours job applies Business Hours policies to identify resources that are running when they should be powered off, reports them, and can start and stop them on the schedule that you specify. Running only during specified business hours can significantly reduce your cloud spend.
   - A cloud account with a service account and associated datacenters is required.
   - Role required: Cloud Insights Admin (sn_clin_core.insights_admin)
   - You can create as many policies as needed.
   - When you deactivate a policy, the resources that met the policy criteria might match a different policy (the matching policy with lowest run order) and therefore move to another schedule. In this case, the system generates a new change request. If a resource no longer meets any policy, the system attempts to power on the resource using the specified Power-on flow setting.

1. Navigate to Cloud Insights > Cost Optimization > Business Hours.
2. On the **Policies** tab, click **New**. Alternatively, edit an existing policy by clicking a policy card or a policy name in the list.

3. On the **Set Policy Properties** page, configure the general settings and the actions that the policy performs on resources that match the policy (meet the policy criteria).

**General section on the Set Policy Properties page**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>The option to apply the policy. Select the <strong>Active</strong> check box to execute Business Hours analysis whenever billing data is updated.</td>
</tr>
<tr>
<td>Policy name / Description</td>
<td>A unique and descriptive name and description for the policy.</td>
</tr>
<tr>
<td>Run order</td>
<td>The order to apply the policies. Each policy must have a unique value. Policies are applied low-to-high. If the condition tag values of a resource match multiple policies, the system matches the resource with the policy with the lowest run order.</td>
</tr>
</tbody>
</table>

**Note:** Run order values for Unassigned Resources policies and Business Hours policies do not interfere with each other.
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy type</td>
<td>The action to take on each resource that matches the policy.</td>
</tr>
<tr>
<td></td>
<td>Auto-apply:</td>
</tr>
<tr>
<td></td>
<td>· Generate a recommendation to apply the specified business hours and add the resource to Business Hours reports.</td>
</tr>
<tr>
<td></td>
<td>· Generate and auto-approve a change request for the assignment group.</td>
</tr>
<tr>
<td></td>
<td>· Apply the Business Hours schedule to the resource.</td>
</tr>
<tr>
<td></td>
<td>Auto-apply after approval:</td>
</tr>
<tr>
<td></td>
<td>· Generate a recommendation to apply the specified business hours and add the resource to Business Hours reports.</td>
</tr>
<tr>
<td></td>
<td>· Generate a change request for members of the assignment group.</td>
</tr>
<tr>
<td></td>
<td>· Add the resource to Business Hours reports.</td>
</tr>
<tr>
<td></td>
<td>· Any member of the group with the ITIL role can approve the schedule (through the change request).</td>
</tr>
<tr>
<td></td>
<td>· When approved, apply the Business Hours schedule to the resource.</td>
</tr>
<tr>
<td></td>
<td>Report-only:</td>
</tr>
<tr>
<td></td>
<td>· Generate a recommendation to apply the specified business hours.</td>
</tr>
<tr>
<td></td>
<td>· Add the resource to Business Hours reports.</td>
</tr>
<tr>
<td>Change template</td>
<td>For the Auto-apply policy type only: Specify the change request template (Standard Change type) to use for Unassigned Resources recommendations.</td>
</tr>
<tr>
<td></td>
<td>If no template appears in the list, you must create one. See Create a change request template.</td>
</tr>
</tbody>
</table>
Schedule and Actions section on the Set Policy Properties page

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business hours schedule</td>
<td>The schedule that specifies the days of the week and times of day that the resource should be powered on.</td>
</tr>
<tr>
<td></td>
<td>Select a schedule from the list or define a schedule by navigating to Cloud Insights &gt; Global Settings &gt; Schedules. For more information, see Define a schedule.</td>
</tr>
<tr>
<td></td>
<td>If you update the time, day, or frequency of a schedule, then the system generates new change requests for the new schedule within 24 hours.</td>
</tr>
<tr>
<td>Power-on flow / Power-off flow</td>
<td>Auto-apply and Auto-apply after approval policy types only. The flows that perform the on/off operations on the resources that match the policy.</td>
</tr>
</tbody>
</table>

4. Click **Next** to move to the **Set Policy Criteria** page.
   A resource matches the policy if the criteria on the **Set Policy Criteria** page are met.

5. In the Service Accounts section, specify the service accounts to apply the policy to.
   a) Select the provider.
   b) Move the required service accounts to the **Selected** list.

6. In the Resources section, specify the conditions (combinations of tag values) that define which resources match the policy.
   - The values come from the Key Value (cmdb_key_values) table.
   - A resource matches the policy if the criteria on the **Set Policy Criteria** page are met.
   - The following example has two conditions. The first condition consists of two required subconditions.
7. Click Save.

The policy appears as a card on the Policies tab. See Viewing Business Hours policies for additional information on the cards.

When the Discovery and billing data update executions finish, the system applies active policies to identify matching resources and then performs the policy actions on the resources.

Viewing Business Hours policies
When you create a Business Hours policy, the policy settings are represented on a card or in a list on the Business Hours Policies tab.

Opening the Business Hours Policies page

Use one of the following methods to access the page:

- Navigate to Cloud Insights > Cost Optimization > Business Hours.
- Navigate to Cloud Insights > Home. In the Business Hours section, click View Details.

Click the Policies tab.

**Note:** When a policy is created or updated with changes other than to its name or description, a notification appears on the tab. Click the Reapply Policies button to apply the changed policy and recalculate the reported data.
Policy cards and lists

- Click the view icon to switch between card view and list view.
- The list view displays additional information like the number of resources in each approval state and creation information for the policy.
- Changes to the name or description of a policy do not change the Last Updated timestamp of a policy.

1. Name.
   - Expected savings and schedules and status of change requests.
   - Run order for the policy.

   **Note:** Run order values for Unassigned Resources policies and Business Hours policies do not interfere with each other.

2. Policy type.
   - Auto: Auto-apply or Auto-apply after approval.
   - Report: Report-only.

3. Hold the cursor here to perform an action:
   - View Report.
   - Clone the policy.
   - Delete the policy.

4. Click anywhere to edit.

5. Active / Not Active.

Viewing the overall state of business hours usage
The Business Hours **Overview** tab identifies the top five opportunities to improve business-hours usage and costs and the number of resources in each state of adjustment of business-hours.
Accessing the Business Hours Overview page

Use one of the following methods to access the page:

- Navigate to Cloud Insights > Cost Optimization > Business Hours.
- Navigate to Cloud Insights > Home. In the Business Hours section, click View Details.

Click the Overview tab.

Working with Business Hours usage data

- All data is auto-updated so there are no settings to configure.
- Data appears on reports only when at least one policy is active.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the Service Account filter.
- Each update of billing data triggers the Business Hours and Rightsizing jobs to analyze spend and usage data and to update the actionable recommendations in reports.
- Because updates to a policy can change the reported data, click Re-Evaluate Policies whenever you have updated a policy. The button is enabled only if all active policies are valid or if any recommendations are still pending.
- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.

Business Hours Overview page

The sections on the Overview tab show the savings that are possible when you implement Business Hours policies.

In the example shown in the figure, the filter settings limit the reports to show data only for the service account named Billing 884788 and only for resources that match the QualityEng policy.
OVERALL COST SAVINGS

Last data download: 2019-11-06 00:12:13

Top 5 business hours savings opportunities

<table>
<thead>
<tr>
<th>Configuration Item</th>
<th>Account</th>
<th>Owner</th>
<th>Monthly Savings</th>
<th>Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Front-end003</td>
<td>DiscoverWS</td>
<td>SalesDev</td>
<td>$424.04</td>
<td>Shopping</td>
</tr>
<tr>
<td>Calcs546</td>
<td>Billing84788</td>
<td>Dev</td>
<td>$542.88</td>
<td>DevVMs</td>
</tr>
<tr>
<td>MonthEndRollup</td>
<td>BillingFinance</td>
<td>Finance</td>
<td>$486.32</td>
<td>FinanceVMs</td>
</tr>
</tbody>
</table>

Progress on business hours savings

$9,160 Last 30-day spend

$1832 (20%) Next 30-day total potential savings

CURRENT STATUS SAVINGS AND DETAILS

Resources scheduled for business hours adjustment

17 Resources

$916 (10%) Next 30-day savings after adjustment

Resources not yet scheduled for business hours adjustment

12 Resources

$916 (10%) Next 30-day savings if adjusted

Declined business hours adjustments

0 Resources

$0 (0%) Declined business hours adjustments

Failed business hours adjustments

0 Resources

$0 (0%) Failed business hours adjustments
Reports on the Business Hours Overview tab

Overall Cost Savings section

- Top 5 business-hours savings opportunities: The five resources that generate the most cost by running outside their specified business hours. Click See All to view details on the Business Hours Resources page. See Managing your business hours processes for more information.
- Your spend and savings from resources in policies: Your total spend over the last 30 days plus your potential savings if the recommended business-hours adjustments are made.

Current status savings and details section

- Resources scheduled for business hour adjustment: The number of resources for which change requests will be generated during the next execution, plus your potential savings for the next 30 days if the recommended business hours adjustments are made. Click See All to view details on the Scheduled Resources tab. See Managing your business hours processes for more information.
- Resources not yet scheduled for business hour adjustment: The number of resources for which change requests are not yet approved plus your potential savings for the next 30 days if the recommended business hours adjustments are made. Click See All to view details on the Pending Resources tab on the Business Hours Resources page. See Managing your business hours processes for more information about working on the Business Hours tab.
- Declined business hour adjustments: The number of resources for which the change requests were rejected, plus the cost of non-business hours operation for the resources over the next 30 days. Click See All to view details on the Declined Resources tab on the Business Hours Resources page. See Managing your business hours processes for more information on working on the Business Hours tab.
- Failed business hour adjustments: The number of resources for which the adjustments failed, plus the cost of non-business hours operation for the affected resources over the next 30 days. Click See All to view details on the Failed Resources tab on the Business Hours Resources page. See Managing your business hours processes for more information on working on the Business Hours tab.
- Resources that match Report-only policies: The number of resources for which no action is being taken plus your potential savings for the next 30 days if business hour adjustments are made on the affected resources. Click See All to view details on the Report-Only Resources on the Business Hours Resources page. See Managing your business hours processes for more information about working on the Business Hours tab.

Managing your business hours processes

Use the information in the Business Hours Resources tab to assess and manage your ongoing efforts to ensure that resources run only when they should run.

Opening the Resources page

Use one of the following methods to access the page:

- Navigate to Cloud Insights > Cost Optimization > Business Hours.
- Navigate to Cloud Insights > Home. In the Business Hours section, click View Details.

Click the Resources tab.
Tabs on the Business Hours Resources page

Because updates to a policy can change the reported data, click **Re-Evaluate Policies** whenever you have updated a policy. The button is enabled only if all active policies are valid or if any recommendations are still pending.

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pending Resources</td>
<td>Resources that are tagged to have business hours enforced, but the change request has not yet been approved.</td>
</tr>
<tr>
<td>Declined Resources</td>
<td>Resources for which owners (or other approver) did not want to enforce business hours (rejected the change request).</td>
</tr>
<tr>
<td>Scheduled Resources</td>
<td>Resources that are tagged to have business hours enforced.</td>
</tr>
<tr>
<td>Report-only</td>
<td>Resources that match Report-only policies.</td>
</tr>
<tr>
<td>Failed Resources</td>
<td>Resources for which the attempt to enforce business hours failed.</td>
</tr>
<tr>
<td>Excluded Resources</td>
<td>Resources that are configured not to be considered for business hours.</td>
</tr>
</tbody>
</table>

Values on the tabs

The following values appear on the **Pending**, **Declined**, **Scheduled**, and **Failed** tabs:

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration item</td>
<td>Unique identifier of the resource.</td>
</tr>
<tr>
<td>Policy</td>
<td>The Business Hours policy that the resource matches.</td>
</tr>
<tr>
<td>Column</td>
<td>Value</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Policy type</td>
<td>The type of Business Hours policy that the resource matches.</td>
</tr>
<tr>
<td>State</td>
<td>For resources that match an <strong>Auto-apply after approval</strong> policy type, status of the change request.</td>
</tr>
<tr>
<td>Change request</td>
<td></td>
</tr>
<tr>
<td>Account</td>
<td>The Service account that includes the resource. A Service account represents a group of related regions in your provider account.</td>
</tr>
<tr>
<td>Owned by</td>
<td>Owner of the resource.</td>
</tr>
<tr>
<td>Failure type</td>
<td>Failed Resources tab only. Reason that the change request was not generated or the specified business hours were not applied to the resource.</td>
</tr>
<tr>
<td>Failure details</td>
<td>Failed Resources tab only. Details that describe the Failure type.</td>
</tr>
</tbody>
</table>

**Rightsizing: Implementing cost-effective sizing for resources**

The Rightsizing feature analyzes resource usage to recommend better sizes for resources that are wasting money by being over-provisioned or underused. A confidence rating and predicted savings support each recommendation to rightsize a resource.

**How rightsizing works**

To generate accurate rightsizing recommendations for resources, Cloud Insights follows this procedure every time billing data is updated:

1. Obtain costs from the updated billing data tables.
2. Collect the CPU and memory usage data for each resource for the preceding 14 days.
3. Obtain rates for resource types and sizes the price sheet data tables.
4. Obtain percentage discount rates from the discount tables and apply the appropriate discounts to the rates on the price sheet.
5. Compare calculated potential costs to actual billed costs and then generate rightsizing recommendations.
6. For each approved change, resize the resource. For a resource in the ON state, stop the resource, resize it, and then restart it. If the attempt to resize fails, perform a rollback, as described in **Rollback on failed Rightsizing attempts**.
7. Update the Rightsizing reports with new recommendations and with approved, rejected, and failed changes.

The recommended memory size is calculated so that the peak usage during the analysis period is no more than 80% of the recommended size. For example, say a resource currently has 16GB and the available sizes are 4GB, 8GB, and 16GB. If the resource used a peak of 3.9GB over the analysis period, then the recommendation would be 8GB.
AWS: Resources that are not considered for rightsizing

Resources with the following AWS attributes are not considered for rightsizing:

- Not in a VPC
- Not backed by an EBS root volume
- Part of Auto-scaling-group
- Does not have enhanced network support
- Virtualization type is not HVM
- Spot instance

Overview: Implementing Rightsizing processes

You follow this general process to define a rightsizing job:

1. On the Rightsizing Recommendations page: Select the resources to rightsize based on the rightsizing recommendations.
2. Add the resources to a Rightsizing job.
3. Specify the time for the job to run and the action to take (auto-rightsize or auto-rightsize after the appropriate user approves the change request).
4. Save the job.

The system immediately generates the change requests. Later, at the scheduled time, the system runs the job to rightsize all approved resources. For change requests that are rejected, failed for some reason, or still pending, you can reschedule the resources into another job.

Ensuring that rightsizing reports and recommendations are current

To ensure that rightsizing reports and recommendations are up-to-date, follow these guidelines:

- Run Discovery on each service account.
- Ensure that the Billing Download job has completed for each provider.
- Ensure that the Price Sheet Download job has completed for each provider.

Confidence levels in rightsizing recommendations

Each recommendation that the system makes to rightsize a resource has an associated confidence level. You consider the confidence level when making the decision of whether to rightsize a resource. Confidence levels reflect the following factors:

- High confidence requires the following conditions:
  - The system has at least 10 days of usage data for the resource.
  - The current and recommended family/generation are identical.

- Medium confidence requires the following conditions:
  - The system has less than 10 days of usage data for the resource.
  - The current and recommended family/generation are identical.

- Low confidence: The current and recommended family/generation are different.
Specify rate discounts to enable accurate pricing for rightsizing recommendations
To generate an accurate rightsizing recommendation, the system analyzes usage data for the last 14 days, obtains prices from the price sheet data tables, and then applies appropriate discounts. To enable the calculations, you specify the provider’s discount rate for each service account.

Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to **Cloud Insights > AWS Settings > Price Discounts**.
2. Click **New** and then fill in the form.

AWS Price Discount form

<table>
<thead>
<tr>
<th>Service account</th>
<th>The service account that the specified discount applies to.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discount (%)</td>
<td>The percentage discount for the specified service account.</td>
</tr>
</tbody>
</table>

3. Repeat the process for each service account.

Configure Rightsizing operations
The Rightsizing feature recommends better sizes for resources that are wasting money by being over-provisioned or underused. Use the Settings page to configure Rightsizing processes and specify the amount of potential rightsizing savings that triggers notifications.

- Run Discovery on each service account.
- Ensure that the Billing Download job has completed for each provider.
- Ensure that the Price Sheet Download job has completed for each provider.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to **Cloud Insights > Cost Optimization > Rightsizing**.
2. On any tab, click **Settings**.
3. On the **General** tab, specify the following settings.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Rightsizing</td>
<td>Option for enabling Rightsizing activities. Clearing the check box disables Rightsizing. Default: Enabled</td>
</tr>
<tr>
<td>After resize, exclude resource for (days)</td>
<td>The number of days to place a resized resource in the Excluded Resources list. The app monitors resized resources for the specified period to verify that the new size is appropriate. When the period expires, the app removes the resource from the list and rightsizing analysis resumes. [<strong>Note:</strong> During this monitoring period, you cannot remove the resource from the Excluded Resources list.]</td>
</tr>
<tr>
<td>Do not recommend if savings are below threshold</td>
<td>The minimum cost over a 30-day period to use when deciding whether to rightsize. For example, a setting of ¥100 means that the app should ignore any rightsizing recommendation that would result in less than ¥100 in savings over 30 days.</td>
</tr>
</tbody>
</table>
Field | Value
--- | ---
Notify when potential rightsizing savings exceed (%) of total spend | The percentage of overall cloud spend of resources that are sized incorrectly that triggers email notification to the users or groups that you specify. The value is calculated using the amount that could be saved if all rightsizing recommendations are followed.

Notify users / Notify groups | The users or groups to notify by email when the spend for incorrectly sized resources exceeds the specified percentage of overall cloud spend. For information on configuring the email, see Create an email template.

4. The default change request templates that the system uses to generate change requests work well in most applications. To specify non-default change templates, follow this procedure.
   a) Select the Advanced check box.
   b) Select the tab for the policy type to configure.
   c) Specify the script include to use for change requests when a resource matches a policy of the specified type.

   Default values:
   - Auto-apply policies (The Assignment group is derived from the Standard Change template and cannot be overridden): CLINRSSStandardChangeRequestUtil
   - Auto-apply after approval policies: CLINRSNormalChangeRequestUtil

   Note: You cannot modify the content of the default scripts.

5. Click Submit.

Define memory metrics (AWS only)
To enable accurate memory usage data for use in generating rightsizing recommendations, you first define memory metrics in your AWS account. You then define a custom memory metric in Cloud Insights.

- Run Discovery on each service account.
- Ensure that the Billing Download job has completed for each provider.
- Ensure that the Price Sheet Download job has completed for each provider.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

AWS does not automatically collect memory metric statistics. On the AWS Management Console, you specify the statistics to collect and push the data to Amazon CloudWatch. Cloud Insights accesses the data through CloudWatch, and verifies that the combination of namespace and metric name is defined correctly. To recommend resources for rightsizing, Cloud Insights analyzes the memory usage data for the custom metric. If no data is returned, the analysis uses the maximum memory of the resource.

1. Navigate to Cloud Insights > Cost Optimization > Rightsizing.
2. Click Settings.
3. On the AWS Memory Metrics subtab on the Memory Metrics tab, click New.
4. Fill in the form and then right-click in the header and select Save.
### Field | Value
--- | ---
Unit | Memory unit of the value.
Namespace | CloudWatch namespace that holds the resource and usage data.
Metric name | Unique name that identifies the custom metric to monitor for rightsizing analysis. Use the name that appears for the metric in CloudWatch.
Resource key dimension | CloudWatch Dimension name that maps to the unique objectID of the VM in your instance, for example, InstanceId.

5. To ensure that the settings that you entered enable the system to access and parse the data correctly, validate the settings against a single region.
   a) Click **Validate**.
   b) In the dialog box, enter the **Service account** and **Region** values to use for validation and then click **Validate**.

After a successful validation, the next cycle of rightsizing analysis uses the custom metric. The custom metric appears in the list on the **AWS Memory Metrics** tab. Error messages for unsuccessful validations recommend corrective actions.

6. Click **Submit** to add the definition to rightsizing analysis.

After a cycle of rightsizing analysis runs, the **Discovered Instances** tab displays the list of resources that meet the custom metric conditions.
1. objectID of the VM
2. Name of the VM
3. Name of the custom metric
4. Name of the CloudWatch dimension

Managing Rightsizing recommendations
Use the Rightsizing Recommendations tab to determine how to manage new recommendations for rightsizing and to manage ongoing rightsizing operations.

Opening the Rightsizing Recommendations page

Use one of the following methods to access the page:

- Navigate to Cloud Insights > Cost Optimization > Rightsizing.
- Navigate to Cloud Insights > Home. In the Rightsizing section, click View Details.

Click the Recommendations tab.
Rightsizing Recommendations page

To take action on a resource, select the recommendation in the list and click one of the actions. For details, see Schedule a resource to be rightsized, Move a resource to a different Rightsizing job, and Exclude a resource from Rightsizing analysis.

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
<th>Available actions on selected recommendations</th>
</tr>
</thead>
</table>
| New Recommendations        | Resources that are candidates for rightsizing. Each update of billing data triggers the Business Hours and Rightsizing jobs to analyze spend and usage data and to update the actionable recommendations in reports. | • Exclude  
• Add to Job  
• Schedule Job |
| Scheduled Recommendations   | Resources that are scheduled in Rightsizing jobs.                                              | • Schedule Job  
• Add to Job  
• Remove from Job |
| Declined Recommendations   | Resources that owners did not want to resize (declined the change request).                    | • Exclude  
• Reschedule in a New Job  
• Reschedule in Another Job |
| Completed Recommendations  | Resources that were rightsized by the Rightsizing job.                                         | None                                                             |
| Failed Recommendations     | Resources for which the Rightsizing operation was not successful. For failed attempts, the Rightsizing operation performs a rollback, as described in Rollback on failed Rightsizing attempts. | • Exclude  
• Reschedule in a New Job  
• Reschedule in Another Job |
| Excluded Resources         | Resources that are configured not to be considered for rightsizing.                           | Remove From Excluded Resources List                              |
## Values on the tabs

The following values appear on the appropriate tabs:

### Values on the New, Scheduled, Declined, Completed, and Failed Recommendations tabs

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration item</td>
<td>Unique identifier of the resource.</td>
</tr>
<tr>
<td>Current size</td>
<td>Size of the resource that was analyzed for rightsizing — the size that you are currently paying for.</td>
</tr>
<tr>
<td>Recommended size</td>
<td>The size that is operationally and cost-wise most appropriate for the resource, based on the analysis of provisioning and usage levels over the last 14 days.</td>
</tr>
<tr>
<td>Override size</td>
<td>Size to use instead of the recommended size. By default, the system uses the <strong>Recommended size</strong> when rightsizing. Specify a different size by specifying an <strong>Override size</strong>.</td>
</tr>
<tr>
<td>Rationale</td>
<td>The reason that the system recommends resizing, based on the analysis of provisioning and usage levels over the last 14 days.</td>
</tr>
<tr>
<td>Monthly savings</td>
<td>Predicted savings if the resource is resized to the <strong>Recommended size</strong>.</td>
</tr>
<tr>
<td>Confidence</td>
<td>The confidence level in a recommendation.</td>
</tr>
<tr>
<td></td>
<td>• High confidence requires the following conditions:</td>
</tr>
<tr>
<td></td>
<td>• The system has at least 10 days of usage data for the resource.</td>
</tr>
<tr>
<td></td>
<td>• The current and recommended family/generation are identical.</td>
</tr>
<tr>
<td></td>
<td>• Medium confidence requires the following conditions:</td>
</tr>
<tr>
<td></td>
<td>• The system has less than 10 days of usage data for the resource.</td>
</tr>
<tr>
<td></td>
<td>• The current and recommended family/generation are identical.</td>
</tr>
<tr>
<td></td>
<td>• Low confidence: The current and recommended family/generation are different.</td>
</tr>
<tr>
<td>State</td>
<td>For resources that match an <strong>Auto-rightsize after approval</strong> policy type, status of the change request.</td>
</tr>
<tr>
<td>Job</td>
<td>Name of the Rightsizing job.</td>
</tr>
<tr>
<td>Planned date</td>
<td>Date that the job is scheduled to run.</td>
</tr>
<tr>
<td>Change request</td>
<td>Change request that is associated with the recommendation.</td>
</tr>
<tr>
<td>Owned by</td>
<td>Owner of the resource.</td>
</tr>
</tbody>
</table>
Schedule a resource to be rightsized

To rightsize a resource, add it to a Rightsizing job and specify when the job should run. You can also move resources out of one scheduled job into a new job.

- Run Discovery on each service account.
- Ensure that the Billing Download job has completed for each provider.
- Ensure that the Price Sheet Download job has completed for each provider.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to **Cloud Insights > Cost Optimization > Rightsizing**.
2. Click the **Recommendations** tab.
3. In the list on the appropriate tab, select the check boxes for the resources to rightszie.
   - To create a Rightsizing job for resources that are new candidates for rightsizing: On the **New Recommendations** tab, select the resources and then click **Schedule Job**.
   - To create a new job for resources that had previously been scheduled but were not resized: On the **Declined Recommendations** tab or the **Failed Recommendations** tab, select the resources and then click **Reschedule in a New Job**.
4. Fill in the dialog box and then click **Schedule**.

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignment group</td>
<td>Assignment group for the resource.</td>
</tr>
<tr>
<td>Account</td>
<td>The Service account that includes the resource. A Service account represents a group of related regions in your provider account.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job name</td>
<td>Unique and descriptive name for the Rightsizing job.</td>
</tr>
<tr>
<td>Scheduled for</td>
<td>Date and time to run the Rightsizing job.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Policy type      | **Auto-rightsize after approval:** Generates a change request for a member of the assignment group to resize the resource and continues to list the resource in the appropriate Rightsizing reports.  
- Until a member of the assignment group approves/rejects the change, the resource state is set to Pending, the resource remains listed on reports, and no other action is taken.  
- If a member of the assignment group approves the request, the state is set to Approved and the resource is added to the list of resources that the job will resize.  
- If a member of the assignment group rejects the request, the resource state is set to Declined, the resource remains listed on reports, and no action is taken.  
**Auto-rightsize:** Generates and auto-approves a change request to resize the resource. The system also sends the approved change request to the assignment group. The resource is added to the list of resources that the job will resize.  
- For the **Change template** field, specify the change request template (Standard Change type) to use for Auto-rightsize recommendations.  
- If no template appears in the list, you must create one. See [Create a change request template](#). |

The Rightsizing job is created and will run at the scheduled time.

Move a resource to a different Rightsizing job

You can add resources to a currently scheduled job, move resources from one job to another, or remove resources from a job.

- Run Discovery on each service account.
- Ensure that the Billing Download job has completed for each provider.
- Ensure that the Price Sheet Download job has completed for each provider.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to **Cloud Insights > Cost Optimization > Rightsizing**.
2. Click the **Recommendations** tab.
3. In the list on the appropriate tab, select the check boxes for the resources that are recommended for rightsizing and then specify the action to take.
   - To add the selected resources to an already scheduled job, click **Add to Job**. In the dialog box, select the job and then click **Confirm**. The resources are added to the job.
   - To add resources that have failed rightsizing or have been previously declined for rightsizing to an already-scheduled job, click **Reschedule in Another Job**. In the dialog box, select the job and then click **Confirm**. The resources are added to the job.
To remove the selected resources from a job that is scheduled to run, click **Remove from Job**. In the dialog box, select the job and then click **Confirm**. The resources are removed from the job.

Exclude a resource from Rightsizing analysis
To ensure that a particular resource is not considered or recommended for rightsizing, add the resource to the Excluded Resource list. The system continues to collect cost and usage data but does not analyze the resource data for rightsizing.

- Run Discovery on each service account.
- Ensure that the Billing Download job has completed for each provider.
- Ensure that the Price Sheet Download job has completed for each provider.
- Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to **Cloud Insights > Cost Optimization > Rightsizing**.
2. On the appropriate tab on the **Recommendations** tab, select the check boxes for the resources and then click **Exclude**.
   The resources are added to the Excluded Resources list. To remove a resource from the list, select it and then click **Remove From Excluded Resources List**.

Rollback on failed Rightsizing attempts
If a Rightsizing action fails, the system immediately performs a rollback to return the resource to its original size, restarts the resource if needed, updates the change request with full details, and updates Rightsizing report data.

In a successful Rightsizing job, the system resizes each resource that has an approved change request. If a resource is in the ON state, the process stops the resource, resizes (modifies) it, and then restarts it. If a resource is in the OFF state, the process resizes the resource but does not start it.

A resource might fail to restart, for example, if the new size would exceed the quota limit for a resource type or for a constrained availability zone.

A Rightsizing job proceeds in batches of resources, grouped by provider/service account/region. If any modified resource in a batch fails to restart, then each resource in the batch is rolled back to its original size and then restarted. The system then updates the change requests for the resources and sets the **Rightsizing recommendation** status on Rightsizing reports to **Failed**.

Resources in a batch that were in the OFF state are not rolled back or marked as failed.

Viewing the overall state of Rightsizing efforts
The Rightsizing **Overview** tab highlights the resources that waste the largest portion of your cloud spend. It also includes reports that show how many resources are scheduled for rightsizing and how many additional resources might benefit from rightsizing.

Opening the Rightsizing Overview page
Use one of the following methods to access the page:

- Navigate to **Cloud Insights > Cost Optimization > Rightsizing**.
- Navigate to **Cloud Insights > Home**. In the Rightsizing section, click **View Details**.

Click the **Overview** tab.
Working with the data

- All data is auto-updated so there are no settings to configure.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the Service Account filter.
- All reports present money values rounded and in reasonable increments. Numbers greater than 10000 are shown as factors of one thousand (k) or one million (m) with two trailing decimal places. For example, €123542 appears as €123.54k and ¥1234567 appears as ¥1.23m.
- Each update of billing data triggers the Business Hours and Rightsizing jobs to analyze spend and usage data and to update the actionable recommendations in reports.

Rightsizing Overview page

The sections on the Overview tab show how much you are spending on underused or over-provisioned resources and your potential savings if the resources are rightsized.

In the example shown in the figure, the filter settings limit the reports to show data only for the service account named Billing 884788 and only for recommendations with confidence levels of High or Medium.
**RECOMMENDATIONS AND SAVINGS PREDICTIONS**

<table>
<thead>
<tr>
<th>Configuration Item</th>
<th>Current Size</th>
<th>Recommended Size</th>
<th>Monthly Savings</th>
<th>Provider</th>
<th>Service Account</th>
<th>Owner</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>d-00459523300877</td>
<td>t3.xlarge</td>
<td>t3.xlarge</td>
<td>$171.15</td>
<td>AWS</td>
<td>Billing 8847880</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>d-000940459223</td>
<td>t2.xlarge</td>
<td>t3.large</td>
<td>$136.05</td>
<td>AWS</td>
<td>Billing 8847880</td>
<td>Low</td>
<td></td>
</tr>
<tr>
<td>d-09823023491092</td>
<td>t3.xlarge</td>
<td>t3.large</td>
<td>$100.44</td>
<td>AWS</td>
<td>Billing 8847880</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>d-0992510982-023</td>
<td>t3.xlarge</td>
<td>t3.large</td>
<td>$85.58</td>
<td>AWS</td>
<td>Billing 8847880</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>d-085240810982</td>
<td>t2.large</td>
<td>t3.medium</td>
<td>$71.59</td>
<td>AWS</td>
<td>Billing 8847880</td>
<td>Low</td>
<td></td>
</tr>
</tbody>
</table>

**Your Spend and Savings from Rightsize Recommendations**

- **$8,947** Last 30-day spend
- **$2,509 (28%)** Next 30-day total potential savings

**RESOURCES AND SAVINGS OVERVIEW**

<table>
<thead>
<tr>
<th>Resources Scheduled to be Rightsize</th>
<th>2</th>
<th>$105 (1%)</th>
<th>Next 30-day savings after rightsize</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resources</td>
<td>176</td>
<td>$2,404 (27%)</td>
<td>Next 30-day savings if rightized</td>
</tr>
<tr>
<td>Resources Declined from Rightsize Change Requests</td>
<td>0</td>
<td>$0 (0%)</td>
<td>Next 30-day savings if rightized</td>
</tr>
</tbody>
</table>
Recommendations and Savings Predictions

- Top 5 Recommendations: The five underused or over-provisioned resources that cost you the most. The columns provide details that help you to determine whether to rightsize a particular resource.
- Your Spend and Savings from Rightsizing Recommendations: Amount that you spent on resources that are good candidates for rightsizing and the predicted savings over the next 30 days if the resources are resized to the recommended levels.

Resources and Savings Overview

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resources scheduled to be rightsized</td>
<td>Number of resources that are included in scheduled Rightsizing jobs and the predicted savings if all resources are resized to the recommended levels. Click <strong>See All</strong> to view the list of resources and associated details.</td>
</tr>
<tr>
<td>Resources not yet scheduled to be rightsized</td>
<td>Number of resources recommended for rightsizing that have not yet been added to a Rightsizing job and the predicted savings if all listed resources are resized to the recommended levels. Click <strong>See All</strong> to view the list of resources and associated details.</td>
</tr>
<tr>
<td>Resources declined from rightsizing change requests</td>
<td>Number of resources where the change request for rightsizing was rejected. Click <strong>See All</strong> to view the list of declined recommendations.</td>
</tr>
</tbody>
</table>

Viewing scheduled Rightsizing jobs
View and manage upcoming rightsizing jobs on the Rightsizing **Jobs** tab.

Opening the Rightsizing Jobs page

Use one of the following methods to access the page:
- Navigate to **Cloud Insights > Cost Optimization > Rightsizing**.
- Navigate to **Cloud Insights > Home**. In the Rightsizing section, click **View Details**.

Click the **Jobs** tab.

Rightsizing Scheduled Jobs page

On the **Jobs** page, select the **Scheduled Jobs** tab.

The following values appear on the tabs. Click a job name to view the job record in the form view.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the scheduled job. Typically auto-generated.</td>
</tr>
<tr>
<td>Planned date</td>
<td>Time that the job will execute.</td>
</tr>
</tbody>
</table>
Viewing completed Rightsizing jobs
When a rightsizing job finishes, the job is listed on the Rightsizing Completed Jobs tab.

Opening the Rightsizing Jobs page
Use one of the following methods to access the page:

- Navigate to Cloud Insights > Cost Optimization > Rightsizing.
- Navigate to Cloud Insights > Home. In the Rightsizing section, click View Details.

Click the Jobs tab.

Rightsizing Completed Jobs page
On the Jobs tab, select the Completed Jobs tab.

The following values appear on the tabs. Click a job name to view the job record in the form view.

### Field | Value
--- | ---
Name | Name of the scheduled job. Typically auto-generated.
Planned date | Time that the job will execute.
Job Status | The status of job execution.
Created by | Name of the admin who created the job.

Identifying unassigned resources to ensure accurate reporting
Unassigned Resources policies help you to identify the resources that are not associated with an assignment group and to assign them appropriately. When a resource is assigned to the correct group, the resource can be appropriately governed even as it goes through stages such as patching, upgrading, and reconfiguring.

How Unassigned Resources policies work
For cloud billing information to be accurate, it must include all resources in your managed cloud infrastructure. An Unassigned Resources policy analyzes all resources to identify resources that are not assigned to an assignment group. The policy can then auto-assign the appropriate assignment group to the resources.

Requirements and limitations
- The Cloud Discovery process triggers Unassigned Resources analysis and updates recommendations in reports.
- The Unassigned Resources feature supports AWS EC2 instances only.
• Terminated, retired, or canceled resources are not considered.
• Data appears on reports only when at least one policy is active. The base system includes a default Unassigned Resources policy. See Viewing Unassigned Resources policies for details.
• When a policy is created or updated with changes other than to its name or description, a notification appears on the tab. Click the Reapply Policies button to apply the changed policy and recalculate the reported data.

Unassigned Resources — Policies tab

To ensure that the information on usage and spending is accurate, Unassigned Resources policies identify resources that do not belong to an assignment group. For such resources, the policy can auto-assign an assignment group to a resource, require a member of the assignment group to approve the assignment, or add the resource to a report and take no further action.

You create and manage Unassigned Resources policies on the Policies tab. See Define or update an Unassigned Resources policy.

Unassigned Resources — Resources tab

Use the Resources tab on the Unassigned Resources page to assess and manage your ongoing efforts to ensure that all resources are associated with an assignment group. Accurately assigned resources gives you greater understanding of and control over your cloud spend.

Configure Unassigned Resources operations

To ensure that the billing data used by Cloud Insights is accurate, the Unassigned Resources feature identifies resources that do not belong to an assignment group. You can choose to use or to disable Unassigned Resources features, and can specify the percentage of unassigned resources that triggers a notification.

• For each provider, run Discovery on each service account.
• Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

1. Navigate to Cloud Insights > Spend Optimization > Unassigned Resources.
2. On any tab, click Settings and then configure the following settings as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Unassigned Resources</td>
<td>Option for enabling the Unassigned Resources feature. All active policies are applied when data becomes available. Deselect the check box to disable the Unassigned Resources feature. No policies are applied regardless of the Active setting for a policy.</td>
</tr>
<tr>
<td>Notify when unassigned resources exceed (% of total CIs)</td>
<td>Level of unassigned resources as a percentage of all CIs in the CMDB that triggers email notification of the users or groups that you specify. For information on configuring the email, see Create an email template.</td>
</tr>
<tr>
<td>Notify users / Notify groups</td>
<td>Users or groups to notify by email when the percentage of unassigned resources spend exceeds the specified limit.</td>
</tr>
</tbody>
</table>
3. Optional: Specify whether to generate a single change request that applies to all CIs that match a policy (the default setting) or a separate change request for each CI.
   a) Select the Advanced check box.
   b) Select or clear the Single change request check box.

<table>
<thead>
<tr>
<th>Setting of the Single change request check box</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected (default)</td>
<td>The system generates a single change request that applies to all CIs that match the policy. For the Auto-apply after approval policy type, the Maximum CIs per change field appears. Specify the maximum number of resources to associate with a single change request. If the maximum is reached, the system generates a new change request.</td>
</tr>
<tr>
<td>Not selected</td>
<td>The system generates a change request for each CI that matches the policy.</td>
</tr>
</tbody>
</table>

4. Optional: Follow this procedure to create a custom script include that sets attribute values in the change request template.

The default change request templates that the system uses to generate change requests work well in most applications. Some attributes of the change request template have static/constant values that cannot be changed. Some attributes (for example, risk and work_notes) have dynamic values that you can set using a script include.

   a) Select the Advanced check box.
   b) Select the tab for the policy type to configure.
   c) Specify the script include that sets values in the change template that is used for the change requests for resources that match the policy.

   To change or add values in the template, create a new script include based on one of the default script includes. In the new script include, override the settings in the setValues function. Upgrades are simplified when you create a new script include.

   - Default for Auto-apply policies (The Assignment group is derived from the Standard Change template and cannot be overridden): CLINUnassignedStandardChangeRequestUtil
   - Default for Auto-apply after approval policies: CLINUnassignedNormalChangeRequestUtil

5. Click Submit or Update.

Define or update an Unassigned Resources policy
To ensure that the information on usage and spending is accurate, Unassigned Resources policies identify resources that do not belong to an assignment group. For such resources, the policy can auto-assign an assignment group to a resource, require a member of the assignment group to approve the assignment, or add the resource to a report and take no further action.

   - For each provider, run Discovery on each service account.
   - Role required: Cloud Insights Admin (sn_clin_core.insights_admin)

   - When a policy is created or updated with changes other than to its name or description, a notification appears on the tab. Click the Reapply Policies button to apply the changed policy and recalculate the reported data.
• The **Cloud Insights** application includes an example policy named Default Unassigned Policy.

1. Navigate to **Cloud Insights > Spend Optimization > Unassigned Resources**.
2. On the **Policies** tab, click **New** or edit an existing policy by clicking a policy card or a policy name in the list.
3. On the **Set Policy Properties** tab, configure the general settings and the actions that the policy performs for resources that meet the policy criteria.

**General section of the policy definition**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Active</strong></td>
<td>The option to apply the policy. Select the <strong>Active</strong> check box to apply the policy whenever billing data is updated.</td>
</tr>
<tr>
<td><strong>Policy name / Description</strong></td>
<td>A unique and descriptive name and description for the policy.</td>
</tr>
<tr>
<td><strong>Run order</strong></td>
<td>The order to apply the policies. Policies are applied low-to-high. <strong>Note:</strong> Run order values for Unassigned Resources policies and Business Hours policies do not interfere with each other.</td>
</tr>
</tbody>
</table>
| **Policy type**        | Action to take on resources that match the policy. **Auto-apply:**  
  - Generate a recommendation to assign the specified assignment group and add the resource to Unassigned Resources reports.  
  - Generate and auto-approve a change request for a member of the assignment group. **Auto-apply after approval:**  
  - Generate a recommendation to assign the specified assignment group and add the resource to Unassigned Resources reports.  
  - When the policy is saved, generate a change request for members of the assignment group. Any member of the group with the ITIL role can approve the change.  
  - When a change is approved, the system assigns the resource to the specified assignment group. **Report-only:** Generate a recommendation to assign the specified assignment group and add the resource to assignment group reports. Perform no other action. |
4. Click **Next** to move to the **Set Resource Criteria** page.
   
   A resource matches the policy if the criteria on the **Set Resource Criteria** page are met.

5. In the Service Accounts section, specify the service accounts to apply the policy to.
   
   a) Select the provider.
   
   b) Move the required service accounts to the **Selected** list.

6. In the Resources section, specify the conditions (combinations of tag values) that define which resources match the policy.

   The tag values come from the Key Value (cmdb_key_values) table.

   A policy applies if any of the specified conditions is met. The following example has two conditions. The first condition consists of two required subconditions.
7. Click **Save**.

The policy appears as a card on the **Unassigned Resources > Policies** tab. For more information on the cards, see **Viewing Unassigned Resources policies**.

The Cloud Discovery process triggers Unassigned Resources analysis and updates recommendations in reports.

**Viewing Unassigned Resources policies**

When you create an Unassigned Resources policy, the policy settings are represented on a card or in a list on the Unassigned Resources **Policies** page.

**Opening the Unassigned Resources Policies page**

Use one of the following methods to access the page:

- Navigate to **Cloud Insights > Cost Optimization > Unassigned Resources**.
- Navigate to **Cloud Insights > Home**. In the Unassigned Resources section, click **View Details**.

Click the **Policies** tab.

**Policy cards and lists**

- Click the view icon
  
  ![View Icon]

  to switch between card view and list view.

  ![List View]

  The list view displays additional information like the number of resources in each approval state and creation information for the policy.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
- Changes to the name or description of a policy do not change the Last Updated timestamp of a policy.

1. Name of the policy.
2. Resources that meet policy criteria and assignment status.
3. Run order for the policy.

   **Note:** Run order values for Unassigned Resources policies and Business Hours policies do not interfere with each other.

2. Policy type.
   - Auto: Auto-apply or Auto-apply after approval.
   - Report: Report-only.

3. Hold the cursor here to perform an action:
   - View resources.
   - Clone the policy.
   - Delete the policy.

4. Click anywhere to edit.

5. Active / Inactive.

### The default Unassigned Resources Policy

At least one Unassigned Resources policy must be active for the Unassigned Resources feature to operate. The base system includes a default Unassigned Resources policy with the following settings:

<table>
<thead>
<tr>
<th>Policy property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Default Unassigned Policy</td>
</tr>
</tbody>
</table>
### Viewing overall progress on unassigned resources

The Unassigned Resources **Overview** tab highlights the number of resources that do not have an assignment group and the status of your Unassigned Resources policies. The Cloud Discovery process triggers Unassigned Resources analysis and updates recommendations in reports.

### Opening the Unassigned Resources Overview page

Use one of the following methods to access the page:

- Navigate to **Cloud Insights > Cost Optimization > Unassigned Resources**.
- Navigate to **Cloud Insights > Home**. In the Unassigned Resources section, click **View Details**.

Click the **Overview** tab.

### Working with the data

- All data is auto-updated so there are no settings to configure. The Cloud Discovery process triggers Unassigned Resources analysis and updates recommendations in reports.
- Because updates to a policy can change the reported data, click **Re-Evaluate Policies** whenever you have updated a policy. The button is enabled only if all active policies are valid or if any recommendations are still pending.
- By default, reports display combined data for all providers. Filter settings control the sources of data for the charts and reports. To display targeted data, change a filter setting. For example, to view data only for a particular service account, select the account in the **Service Account** filter.

### Unassigned Resources Overview page

- The sections on the **Overview** tab show how many resources are still unassigned and how many resources have been identified by the policies that you have defined.
• On this page, each See All button opens the Unassigned Resources > Resources page to the appropriate tab. See Managing how resources are associated with assignment groups for information on working on the Resources tab.

• In the following example, the filter settings limit the reports in the Policy Overview section to show data only for the service account named Billing 884788 and only for resources that match the QualityEng policy.
UNASSIGNED RESOURCES OVERVIEW

Unassigned resources in the CMDB

2 (2%)
Unassigned resources of 101 total

POLICY OVERVIEW

Assigned resources

42 Resources
1 Policy

Pending assignments

38 Resources
1 Policy

Declined assignments

1 Resource
1 Policy

Failed assignments

0 Resources
0 Policies

Report only

20 1
Unassigned Resources Overview section

- Unassigned resources in the CMDB: The number and percentage of resources that do not have an assignment group. The report is not affected by filter settings. Click See All to open the Unassigned Resources page to the Unassigned Resources tab.

Policy Overview section

- Assigned resources: The number of resources that matched a policy and were assigned to an assignment group. Click See All to open the Assigned Resources tab.
- Pending assignments: The number of resources that matched a policy and have been flagged for assignment, but the change request has not yet been approved or the action hasn’t been taken. To work on the resources, click See All to open the Pending Resources tab.
- Declined assignments: The number of resources where the approver rejected the change request. To work on the resources, click See All to open the Unassigned Resources page to the Resources > Declined Resources tab.
- Failed assignments: The number of resources that matched a policy but failed to be assigned. To work on the resources, click See All to open the Unassigned Resources page to the Resources > Failed Assignment tab. You can work with the resources on that tab.
- Report-only: The number of resources that matched a Report-only policy type. To work on the resources, click See All to open the Unassigned Resources page to the Resources > Report-only tab.

Managing how resources are associated with assignment groups

Use the Resources tab on the Unassigned Resources page to assess and manage your ongoing efforts to ensure that all resources are associated with an assignment group. Accurately assigned resources gives you greater understanding of and control over your cloud spend.

Opening the Resources page

Use one of the following methods to access the page:

- Navigate to Cloud Insights > Cost Optimization > Unassigned Resources.
- Navigate to Cloud Insights > Home. In the Unassigned Resources section, click View Details.

Click the Resources tab.

Tabs on the Resources page

Because updates to a policy can change the reported data, click Re-Evaluate Policies whenever you have updated a policy. The button is enabled only if all active policies are valid or if any recommendations are still pending.
Select a resource and click **Exclude** to exclude it from reports. See [Exclude a resource from Cloud Insights reports](#).

<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
<th>Available actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pending Assignments</td>
<td>Resources that matched a policy and are scheduled for assignment.</td>
<td>Exclude</td>
</tr>
<tr>
<td>Declined Assignments</td>
<td>Resources that owners did not want to resize (rejected the change request).</td>
<td>Exclude</td>
</tr>
<tr>
<td>Assigned Resources</td>
<td>Resources that were flagged for assignment (by either an Auto-apply or Auto-apply after approval policy) and have been assigned to an assignment group.</td>
<td>Exclude</td>
</tr>
<tr>
<td>Report-only</td>
<td>Resources that matched a Report-only policy.</td>
<td>None</td>
</tr>
<tr>
<td>Failed Assignments</td>
<td>Resources that matched a policy but were not assigned.</td>
<td>Exclude</td>
</tr>
<tr>
<td>Excluded Resources</td>
<td>Resources that are not included in the Unassigned Resources analysis process. To exclude a resource, select it on any tab and then click <strong>Exclude</strong>.</td>
<td>Remove From Excluded Resources List</td>
</tr>
<tr>
<td>Unassigned Resources</td>
<td>Resources that do not currently have an assignment group.</td>
<td>(None)</td>
</tr>
</tbody>
</table>

**Values on the tabs**

The following values appear on the **Pending**, **Declined**, **Assigned**, and **Failed** tabs:
<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration item</td>
<td>Unique CI identifier of the resource.</td>
</tr>
<tr>
<td>Policy</td>
<td>The policy that the resource matched.</td>
</tr>
<tr>
<td>Policy type</td>
<td>The type of policy that the resource matched.</td>
</tr>
<tr>
<td>State</td>
<td>For resources that match an <strong>Auto-apply after approval</strong> policy type, status of the change request.</td>
</tr>
<tr>
<td>Change request</td>
<td>Change request that is associated with the recommendation.</td>
</tr>
</tbody>
</table>
| Account                    | The Service account that includes the resource.  
                                A Service account represents a group of related regions in your provider account. |
| Owned by                   | Owner of the resource.                                                |
| Failure type               | Type of failure.  Failed Assignments tab only.                        |
| Failure details            | Explanation of the failure.  Failed Assignments tab only.             |
Index

A
accessibility features 1392
accessible 1392
action 1566
activate 1627
active change window 1683
active connection information
Solaris 875
Activities page 2509
AD active directory 1497
add a service to a group 286, 1384
add CI 265
Add segment of service 269
Add text and event fields to an alert 1420
additional info 1398, 1439
additional information 1398, 1439, 1491
Additional Information 1438, 1568, 1569, 1668, 1673, 1674, 1674, 1677, 1710, 1710
Additional Information 1610, 1611, 1707
ADM 718
administer 1394
advanced configuration 1657
aggregated alert groups 1591
aggregated connections 312
aggregated edges 312
Aggregated group 1596
aggregation 1596, 1620, 1623, 1626, 1627
AI 1568, 1673, 1674, 1674, 1677, 1696, 1697, 1710, 1710
AIX 1180
alert
impact 1577, 1582, 1617
monitoring 1577, 1582, 1609, 1617
alert action rule 1547, 1555, 1679
alert action rules 1370
alert aggregation 1721
alert aggregation and root cause analysis 1346
alert aggregation and root cause analysis RCA 1357
alert console 1386, 1387, 1591
alert correlation rule 1679
alert group 1707
alert insight 1619, 1696, 1697
Alert Insight 1568, 1673, 1674, 1674, 1677, 1710, 1710
alert intelligence 1619, 1696, 1697
Alert Intelligence in Workspace 1604, 1607
alert management rule designer 1555
alert management rules 1562, 1679
alert messages 1545
alert monitoring 1598, 1604, 1605, 1630
alert point 1707
alert priority 1610, 1611, 1707
alert promotion rule 1909
alert promotion rule blacklist 1910
alert remediation 1557
alert report 1630
alert score 1610, 1611, 1707
alert table 1370
alert workspace 1620
Align versions of customized probes and sensors 665
all alerts 1387
all events 1394
Analyzing
service 300
Application Dependency Mapping 718
Application Dependency Mapping (ADM)
overview 715
requirements
Windows 2000 server 715
application pattern
eexample 1311
application server 1359, 1388
application service
customization 367
defining 197, 203
text point 203
history 321, 327
properties 367
Application service maps 301
application service review 253, 259
arrange services in groups 286, 1384
attach a knowledge base (KB) article 1554
attach article 1554
attach KB 1554, 1696, 1697
audit log 1545
Authentication and data security options 1524
Authentication type 1524
auto close alerts 1390
auto grouping CI 335
auto-grouping CI 335
automated business service 1603
Automated group 1596
AWS platform 1544
Azure
service principal 1959
Azure Activity 1545
Azure Classic Metric Alert 1545
Azure log Alert 1545
Azure platform 1545
Azure service principal 1968

B
baseline, set 321
Basic Auth Credentials 1465, 1493, 1497, 1499
Behaviors
multiple domains 733
benchmark performance 1665
Best Practice
add application services 190
add business services
by importing 377
manually 377
add services 190, 190
by importing 190, 378, 380
create phase 376
en masse 190, 378, 380

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
from load balancer 380
import 380
import business services 377, 378, 380
import services 190
in bulk 190
load business services 377, 378, 380
load services 190
manually 380
multiple 190, 380
phase 376, 377
populate phase 377, 378
potential business service 380
Bi-directional 1486
bi-directional connector 1511
bind 1544
black list blacklist 1413
blacklist 1415
Boundary 259
bulk API 1526
bulk event collector URL 1536, 1539
Business Hours policies 2587
Business rule 1370
business rules 1359, 1656
Business rules installed with Event Management 1359
business service
  defining 199
  entry points 199
  history 327

C

Calculate related tasks 1568, 1673, 1674, 1674, 1677, 1710, 1710
Cascading Style Sheet (CSS) formats 1392
change credentials 1230
change history
  viewing 321, 327
change record 321
change request 1617
change request window 1683
Change user operation 1230
checkout redirect property
  CloudFormation 2120
child services 1589
CI
  adding to map 259
  check dependencies 332
  config files 311
  configuration files 311
  display 367
  in business services 332
  marking as boundary 259
  properties 367
  removing from map 259
Service Mapping
  attributes 311
  name labels 311
  properties 311
  supported by Service Mapping 34, 397
  CI binding 1370
  CI bound to alerts 1392
CI class
  creating 1203
  Service Mapping 1203
CI remediation 1695
CI type
  creating 1203
  Service Mapping 1203
CI types
  discovery schedule for 290
  updating 290
CIM
  parameter 607
CIM Discovery 955
CimIQL
  component tokens 611
  operation tokens 609
  results 614
  syntax 608
  tutorial 613
classification
  configuration item 538
  device 531
  process 542, 880
  script 546
classifiers
  horizontal pattern probe 601
Cloud Insights
  system properties 2549
Cloud Insights admin guide 2551
Cloud Insights getting started 2521
Cloud Insights setup guide 2522
Cloud Management
  API 2404
  billing data 2542
  billing data size 1949, 1977
  billing download 2537
  cloud billing schedule 1948, 1974
  incident 2508
  price sheet 2545, 2547
  service accounts 1969, 2007
Cloud REST Call
  Pattern Designer 250, 276, 276, 1040, 1214, 1214,
  1214, 1214, 1219, 1219, 1219, 1219, 1219, 1227, 1227,
  1227, 1227, 1227, 1230, 1231, 1236, 1237, 1241, 1243,
  1247, 1247, 1251, 1255, 1257, 1261, 1277, 1282, 1284,
  1287, 1289, 1291, 1293, 1295, 1297, 1297, 1297, 1297,
  1299, 1299, 1299, 1301, 1301, 1301, 1303, 1303, 1303,
  1303, 1303, 1305, 1305, 1305, 1305, 1308, 1308, 1308,
  1309, 1311, 1329, 1332, 1342, 1342, 1343, 1343, 1343,
  Cloud REST Call operation 1040, 1261
  cloud service accounts 1950, 1984
CloudFormation
  checkout redirect property 2120
clusters
  display 259
  types 259
CMDB alert groups 1725
CMDB group 1596, 1620, 1623, 1626, 1627
cmdb_ci_cloud_load_balancer table 2513
cmdb_ci_endpoint 1207
cmdb_ci_linux_server 1369
cmdb_ci_network table 2513
create maintenance rule 1683
create or close alerts 1424
create security incident 1568, 1569, 1668, 1673, 1674, 1674, 1677, 1710, 1710
create task 1568, 1569, 1668, 1673, 1674, 1674, 1677, 1710, 1710
creating an SLA configuration 1369
credential 1464, 1498, 1737
credentials
  checking IP service affinity 565
credentials, Service Mapping 118
cross business service impact 1589
custom connector 1520
custom connector definition 1449
custom operation 1106
custom properties 1491
custom script 1659
customizable dashboards 1630
customize
  Service Mapping 343
Customizing
  mapping patterns 1024
D
dashboard 1598, 1631, 1640, 1648, 1652
dashboard view 1602
dashboards 1604
data analysis 1568, 1673, 1674, 1710
data collected 785
database catalog 913
datadog 1548
Datadog 1548
datastores 804
de-duplication 1370
define access to services 287, 1382
define user access to services 287, 1382
defining a pattern

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
domain separation 525
error messages
  no sensor defined, identifying 1022
  probe not found, identifying 1023
  sensor, fixing 1022
  sensor, identifying 1020
ESX servers 798
export patterns 1340
extended capabilities of 678
horizontal discovery log 597
horizontal discovery process flow 392, 394
hyper-v 760, 764
identifiers
  process description 574
identity probes
  configuring 575
IP addresses used 521
JBoss servers
Linux probes and sensors 766
layer 2
  introduction to 836
list of probes 607
load balancers 813
log 595
log retention 599
logs 593
managing large payloads 968
MySQL 925
networks 452
NGINX web server 893
Operating system level virtualization (OSLV)
  Docker introduction 974
Oracle databases 928
Parsing 1297
pattern designer
  creating relationships and references between CIs 1266
pattern log 597
permissions 638, 663
ports and protocols 651
powershell 696
PowerShell probe 616
probe types 389
probes 602, 638, 657, 659, 663, 668, 673, 674
Puppet
  Add the must_sudo parameter to the Puppet probe 898
Puppet software 896
range sets, importing 524
SCCM
  tracking software with 530
SCCM integration 529
schedule 520
sensors 602, 657, 668, 673, 674
serial number types 580
serverless 504
Set SNMP version on the Discovery schedule 529
shell script 875
SNMP MIBs 623
SNMP probe 620
SNMP probe parameters 620
SNMPv3 support 526
software discovery
  clustered Windows applications 880
source name 508, 753
status 581, 584, 587, 589
Status 583
storage 936
storage data collected
  SMI-S and CIM 947
sudo 869
timeline 584
use patterns 601
vCenter probe upgrades 634
Veritas Volume Manager data 946
VMware infrastructure 782
VMware relationships without ESXi server 804
VMware vCenter 782
WebLogic application server 906
WebLogic servers
  Windows probes and sensors 910
WebSphere application server 910
what Discovery can discover 395
Windows servers
  clustered discovery 809
discovery application mapping
  UNIX 875
Discovery behaviors
  load balancing 724
Discovery credentials
  Discovery schedule 290
discovery pattern
  condition 1222
  set precondition 1222
  variables 1228
Discovery patterns
  horizontal discovery 1329
discovery probe
  horizontal pattern probe 616
Discovery range sets 461
discovery schedule
  Service Mapping 290
discovery steps
  define 1211
  manage 1211
Discovery Trap Collector Extension 1550
discovery> 785
dll 1486
Docker
  introduction to 974
  does not display in the Alerts console 1386
domain aware 1656
domain separation
  Discovery 525
Domain Support – Domain Extension Installer plugin
  1359, 1388
drill down 1603
dynamic time 1386

E
e-grep 364
egrep 364
elevated rights 118
em_unique_nodes 1392
e-mail 1370
e-mail notifications
creating 1391
e-entry point
class 1207
type 1207
e-entry points 199
e-errors
Service Mapping 250
ESX resource pools
configuring 801
ESX servers
discovery of 798
eval 1214
event collection to MID Server 1524
event information 1394
event information pane 1413
event input 1413
event integration 1370
event management
service groups 286, 1384
Event Management
activate 1352
additional configuration 1376
alert
active interval 1571
alert maintenance 1688
alert maintenance status 1682
flapping
detection 1572
incidents 1693
properties 1571
rules 1579
Alert and CI remediation 1628
alert binding 1426
alert correlation
rules 1716
alert history 1617
alert monitoring 1597
alert process flow 1679
alerts 1571
Alerts 1681
assign role 1388
bind alerts to CIs 1426
business service
impact calculation 1579
business service group 1388
CI Bind 1438
CI infrastructure 1585
collaborate 1602, 1688
configuration 1376, 1571
configuring event correlation rules 1718
creating an SLA 1370
creating an SLA configuration 1368
dashboards 1597
discovery 1389
e-mail connector instance 1553
event
create manually 1407
rules 1419
event field mappings 1404
event groups 1439
event identifier 1397
impact calculation 1579
Impact Tree 1577
manual service 1609
Manual Services
impact calculation 1579
MID Server)
plugin 1352
POP3 connector instance 1553
rules
bind 1432, 1435, 1437
processing for alerts 1418, 1419
processing for events 1418, 1419
secure connection 1846
service map 1582
SLA configuration
limiting the SLA configuration records returned
1369
technical service 1376
technical services
impact calculation 1579
topology 1582
view correlated alerts 1717
Event Management alert 1438, 1523, 1550, 1596, 1603, 1607, 1683, 1699, 1693
Event Management alert action rules 1570
Event Management alert group 1386
Event Management alert incidents 1693
Event Management alert management rule 1555
Event Management Alerts Console 1605
Event Management architecture 1346
Event Management assign role 1491
Event Management AWS CloudWatch 1544
Event Management Azure 1545
Event Management best practice 1520
Event Management collect events using external event sources 1526, 1541
Event Management configuration 1357
Event Management connector 1453, 1522
Event Management connector instance 1446
Event Management Context extension configuration 1533
Event Management correlated alert groups 1591
Event Management create alert group 1596, 1607
Event Management create alert management rule 1557
Event Management create CI automatically 1398
Event Management create event rule 1411
Event Management custom connector 1500, 1501, 1511
Event Management custom metric connector 1517, 1519
Event Management customize alert content 1420
Event Management Dashboard 1598
Event Management documentation links 1346
Event Management domain separation 1359, 1388
Event Management event 1394, 1523, 1548, 1550
Event Management event field mapping 1401
Event Management event fields 1398, 1439
Event Management event rule 1410
Event Management event threshold 1424
Event Management external event source 1540
Event Management filter 1415
Event Management high availability 1553
Event Management HPOM connector instance 1456
Event Management impact free 1617
Event Management installed components 1360, 1363
Event Management integration 1398, 1439
Event Management JavaScript script 1500, 1511
Event Management MID Server 1842
Event Management MID Server
MID Web Server 1523
Event Management monitored service 1603
Event Management NagiosXI connector instance 1464
Event Management Netcool connector instance 1458
Event Management NNMI connector instance 1462
Event Management OMI connector instance 1453
Event Management Panopta 1547
Event Management purge data 1390, 1390, 1390, 1390
Event Management quick response 1695
Event Management rules bind 1431
Event Management SCOM connector instance 1486, 1486
Event Management SCOM group 1491
Event Management service group 1387
Event Management SNMP traps 1550
Event Management SolarWinds connector instance 1491
Event Management vCenter connector instance 1494
Event Management VMware vRealize Hyperic connector instance 1457
Event Management VMware vRealize Operations 1495
Event Management Zabbix connector instance 1498
Event Management Lifecycle
alert 1439, 1554, 1571, 1572, 1579, 1602, 1668, 1682, 1688, 1693, 1694, 1694
event mapping rules 1370
Event message headers 1524
event rule 1413, 1424
event rule designer 1410
event rule wizard 1410
event rules 1370
event stream notification 1536, 1539
event table 1370
event table cleanup 1390, 1390, 1390, 1390
event table rotation 1390, 1390
Events collected from BMC ProactiveNet Performance Management (BPPM) 1536, 1539
Events collected from BMC TrueSight Operations Management 1536, 1539
evt_mgmt.alert_black_list_fields 1438
evt_mgmt.domain_self_monitoring 1656
evt_mgmt.event_processing_stats_oldest_ready_event_period 1665
evt_mgmt.event_processing_stats_period 1665
evt_mgmt.override_alert_short_description 1438
evt_mgmt.self_health_active property 1655
example: populate a CI with a text file 676
exclude 2548, 2548
execution 1568, 1569, 1673
extensions 1552

external devices 1442

F

field mapping 1547
File type parsing strategy 1297, 1308
filter 1707
Filter table
Conditions 1237
Filter table operation 1237
filter token 610
Find matching URL operation 1236
fix map 247, 265

G

GCP 1083
GCP Label, key-value pairs 1083
GCP service accounts 1900
genericJSON 1442, 1541
Get process operation 1240
Get registry key
Pattern Designer 250, 276, 276, 1040, 1214, 1214,
1214, 1214, 1219, 1219, 1219, 1219, 1227, 1227, 1227,
1230, 1231, 1236, 1237, 1241, 1243, 1247, 1247, 1251,
1255, 1257, 1261, 1277, 1282, 1284, 1287, 1289, 1291,
1293, 1295, 1297, 1297, 1297, 1299, 1299, 1299,
1301, 1301, 1301, 1303, 1303, 1303, 1303, 1305, 1305,
1305, 1305, 1308, 1308, 1309, 1311, 1329, 1332,
1342, 1342, 1342, 1343, 1343, 1343
Get registry key operation 1241
getting started 1346
Golden Gate 1180
group 1591
group CIs 335
group priority 1610, 1707
grouping CIs 335
Guided setup 1357
guidelines 1370, 1520

H

handleMetric 1517, 1519
hardware
Cisco UCS 857
WebSphere DataPower devices 856
Health 1345
health check monitors 1640, 1648
history view 321, 327
horizontal discovery 1257
host less 33
hostless 33
HP OMI 1453
HTTP Get Call
Pattern Designer 250, 276, 276, 1040, 1214, 1214,
1214, 1214, 1219, 1219, 1219, 1219, 1227, 1227, 1227,
1230, 1231, 1236, 1237, 1241, 1243, 1247, 1247, 1251,
1255, 1257, 1261, 1277, 1282, 1284, 1287, 1289, 1291,
1293, 1295, 1297, 1297, 1297, 1299, 1299, 1299,
1301, 1301, 1301, 1303, 1303, 1303, 1303, 1305, 1305,
1305, 1305, 1308, 1308, 1309, 1311, 1329, 1332,
1342, 1342, 1342, 1343, 1343, 1343

manual alert 1596
manual service 1609
Manually create events 1407
Manually send events 1407
Map
Service Mapping 300
map indicators 301
map view
Service Mapping 335
mapping
service mapping 23
Mapping boundary 259
Mapping credentials 76
mapping patterns 1024
maps
Service Mapping 301
Match condition 1245
Match criteria 1245
Match operation 1245
mean time to resolve 1620, 1623, 1626, 1627
measure network performance 1462
merge criteria 1233
Merge table operation 1233
merged connections 312
merged connectors 312
merged edges 312
merging edges
Service Mapping 335
metric collection 1486
MetricCollector 1517, 1519
Microsoft
SQL server clusters
discovering 916
MID server
configuration 374
MID Server
configure default for Service Mapping 72
selection sequence for Discovery 444
Service Mapping 67
MID Server algorithm 72
MID Server collection endpoint 1550
MID Server configuration 1451, 1838
MID Server Operational Intelligence extension 1842
MID Server Operational Metric Manager Context 1842
MID Server pushes events to instance 1552
MID Server selection 72
MID SNMP Trap Listener 1550
MID Web Server 1533, 1536, 1539
migrate 1377, 1380
migrate alert action rule 1570
missing CI 265
mobile devices 1631, 1640, 1648
mobile support 1630
modeling solution 1386
modify order 1610, 1611, 1707
Modify patterns 1332
Modifying
mapping patterns 1024
monitor configuration scripts 1652
monitor network resources 1462
monitoring tools notification 1394
MTTR 1620, 1623, 1626, 1627
multi-factors 1610, 1611, 1707

N
Nagios XI 1464
Nagios XI server 1464, 1465
NagiosX1 1464, 1465
Netflow 357, 359
network 1106
network Discovery
add Discovery range set to a Discovery schedule 461
convert IPs to range sets 459
introduction to 452
schedule for 455
Network Flow Logs 363
network path
Service Mapping 332
network pattern, GCP 1083
Network Performance Monitor (NPM) 1491
Network Pillar pattern, GCP 1083
network resource table 2513
NNMi 1462
node count 1392
noise reduction 30, 268
non-elevated rights 137
non-privileged user 137
non-sudo commands 137
non-super user 137
normalize 1438
occurrence of something of interest 1394
OEM 12c Trap event rule 1552
OOB 1360, 1557
OP5 Monitor 1466
open Dependency Views 332
operating system
support for non-English 374
Operating system level virtualization (OSLV)
Docker
introduction for 974
introduction to 974
operation token 608
Operational Intelligence 1346, 1346, 1909, 1910
Operational Intelligence Metrics 1842
Operational Metrics 1346
Operational status 1655
OperationsManagerDW 1486
operator 1737
Opsview Monitor 1467
Optimization 1923
Oracle
discovery of 928
Oracle Enterprise Manager 1552
Orchestration
credentials for
checking IP service affinity 565
PowerShell probe 616
organize services in groups 286, 1384
OS
  support for non-English 374
other Azure formats 1545
out of the box 1557
Out of the box 1360
output 1566
overview 1346, 1631, 1640, 1648, 1652
overview dashboard module 1630

P

PaaS 1392
Paessler PRTG Network Monitor 1469
Parameter Token 611
parent alert 1596, 1607
Parse command output
  Pattern Designer 250, 276, 276, 1040, 1214, 1214,
  1214, 1214, 1219, 1219, 1219, 1219, 1227, 1227, 1227,
  1227, 1230, 1231, 1236, 1237, 1241, 1243, 1247, 1247, 1251,
  1255, 1257, 1261, 1277, 1282, 1284, 1287, 1289, 1291,
  1293, 1295, 1297, 1297, 1297, 1297, 1299, 1299, 1299,
  1301, 1301, 1301, 1303, 1303, 1303, 1303, 1305, 1305,
  1305, 1305, 1308, 1308, 1308, 1309, 1311, 1329, 1332,
  1342, 1342, 1342, 1343, 1343, 1343
Parse command output operation 1247
parse event messages 1523
parse file
  traffic-based connection 1334
Parse file 1251
Parse file operation 1251
Parse URL operation 1253
Parse variable operation 1255
Parsing strategies 1297
Patten Designer
  add column to table 1289
  append tables 1292
  match 1245
  merge table 1233
  union tables 1292
pattern
  connection 1231
  connection section 1334
  value 1214
  variable 1214
Pattern
  HTTP 1257
  PaaS 1040, 1261
pattern definition
  example 1311
pattern designer
  creating relationships and references between CIs 1266
  horizontal discovery 599
Pattern Designer
  change user 1230
  Command Prompt 1227
  Create connection operation 1231
  Debug mode 1219
  Discovery 1297, 1299, 1301, 1303, 1305, 1308, 1342,
  1343
  extension sections 1332
Filter table 1237
Find matching URL 1236
GA 276
generic application 276
Parse file 1251
Parse variable 1255
parsing strategy 1297, 1299, 1301, 1303, 1305, 1308
pattern example 1311
related CIs 1329
Run command 1219
Run Command 1227
Search Assistant 1219
Service Mapping 1297, 1299, 1301, 1303, 1305,
  1308, 1342, 1343
Set parameter value 1282
Shared Library 1277
shortcuts 1214
SNMP query 1287
Test 1219
test command 1227
transform table 1289
troubleshooting patterns 250
UI 1214
Unchange user 1291
upload files 1284
user interface 1214
WMI query 1295
working with 1214
pattern version 1342, 1343
payload URL 1526
payloads
  managing size for Linux or Solaris 968
PCF 1185
performance 1370, 1665
Performance Analytics - Content Pack 1630
permission 1737
Pivotal 1185
place alert in maintenance 1688
place CI into maintenance 1688
plan business services 190, 376, 377, 378, 380, 380
planning 1370
Planning
  service 300
platform upgrade 1358
populate 1438, 1544
Port Probes 648
PowerShell 694
PowerShell probe
  creating 618
predictive intelligence 1619
preferred 1370
premium 1630
primary 1596, 1620, 1623, 1626, 1627
primary alert 1596, 1607, 1718
priority 1605
Priority Breakdown 1568, 1610, 1611, 1668, 1673, 1674,
  1674, 1677, 1707, 1710, 1710
priority calculation 1611
priority computation 1611
Priority group 1605
privileged user 118
probe
  WMIRunner 647
probes
  horizontal pattern probe 601
  upgrades for vCenter 634
process
  classification 542, 546, 880
  process alarms as events 1544
  process event messages 1545
  process handler
    create 547
Processing Notes 1394
progress 1625
promote anomaly alerts 1909, 1910
propagate 1589
Properties installed with Event Management 1359
Properties Token 611
provided with the base system 1562
Provided with the base system 1360
PRTG 1469
pull 1442
pull connector 1449, 1466, 1467, 1469, 1500, 1511, 1517, 1519
pull event information 1550
pull operation 1522
Puppet
  Add the must_sudo parameter to the Puppet probe 898
  automation software discovery 896
purge alerts 1390
push 1442
push connector 1544
push event information 1523, 1550
push event messages 1533, 1536, 1539
push external events from monitoring tools 1540
push operation 1523, 1524
put file 1284
Q
QBS 1376
query 608
Query Delimiter Token 611
quick discovery 439
quick IP range
  creating 522
R
range sets 459
raw metric information 1517, 1519
RawMetric 1517, 1519
RCA 1346
ready 1386
reclassifying
  ServiceNow Discovery 560
workstation
  workstations 560
recommendation 1520
regex 1413
regular alert 1909
Regular expression parsing strategy 1303
reject message 259
Related CIs
discovery 1329
remediation 1370, 1555, 1557, 1570
replace CI 265
reports 1630, 1734
resolution 1566, 1619
resolve 1568, 1569, 1673, 1674, 1674, 1677, 1710
result 609
resume 247
retired 1386
retrievekpi 1517, 1519
role 1737
roles 1359
Roles installed with Event Management 1359
root cause 1568, 1673, 1674, 1674, 1677, 1710
root cause analysis 1721
S
sa.qbs.max_num_of_cis 1369, 1376
Save and upgrade 1411
scale environment 1370
SCCM
  integration with Discovery 529
  tracking software with 530
schedule
  application 290
  applicative CI 290
scheduled job 1657, 1665
scheduled jobs 1359
Scheduled jobs installed with Event Management 1359
scorecard 1640
SCP Relay Probe 618
script
  on classification 546
  script include incidents 1693
  Script include to transform collected events 1524
  script includes 1359
  Script includes installed with Event Management 1359
  script instructions 1522, 1524
  scripted integration 1540
  secondary 1596, 1620, 1623, 1626, 1627
  secondary alert 1596, 1607, 1718
  Secure Connection 1524
  security incidents 1693
  segment of business service 269
  self monitoring 1652
  self-health 1652
  self-health application service 1662
  self-health manual service 1662
  self-health monitoring 1656
  self-monitoring 1370
service
  create 190
  define 190
  indicators 335
service accounts 1942
Service Analytics documentation links 1346
service health monitoring 1598

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
service mapping
service groups 286, 1384
Service Mapping
508 342
   accessibility mode 342
   activating 64
   activation 64
add application services 195, 195
add business services 195
add business services manually 380
application permissions 76
application rights 76
application service 247, 282
AWS logs 30
behavior 186
Best Practice 376, 377, 378, 380, 380
business service 190, 247, 265, 282, 375
by importing 195
change request 263
changes 321, 327, 751
check connectivity 383
commands 118, 137, 183
compare configuration files 321, 327, 751
configuration 67, 343, 367
configuration files 321, 327, 751
Connect to existing service 269
create connection section 1334
Create new service 269
Create new service from here 269
create pattern 1024
creating a traffic based discovery rule 343
creating a traffic-based discovery rule 343
credentials 76
Credentials 62
CSV file 375
customization 367, 1024, 1339
customize map view 335
customizing maps
   merging connector lines 335
data collection 25
Debug mode 1219
define business service 281
discover business services 191
discovery flow 25
discovery message 240
discovery patterns 1024, 1224
Discovery patterns 1247
discovery rule 343
discovery schedules 290
domain separation 32, 67, 197
en masse 191, 195
ten point 1207
ten point connectivity 383
errors 240
export patterns 1340
F5 186
finalizing patterns 1339
fine-tune business service 263
fix application service 263
fix application services 233
flow 30
GA 276
Generic application 276
Get started 62
Guided Deployment 381, 383, 385, 387
ignore errors 240
implement comments 263
import business services 195, 375
import entry points 375
import from CSV 195, 378
in bulk 191, 195
Installation 62
IP address connectivity 383
IT service 265
keyboard 342
keyboard shortcuts 342
load application services 195
load balancer 186
Issof 30
map 240, 247, 247, 247, 265, 265
map application services 191, 191, 191
map multiple business services 191
Mapping
   approve 259
   Checking results 253, 259
   Discovery messages 253, 259
   Errors 253, 259
   Results 253, 259
   review 259
mapping flow 25
mapping patterns 1024
maps 301
MID Server
   IP range 72
   MID Server algorithm 67
   MID Server choice 72
   MID Server pool 72
   MID Server selection 67
   MID Server selection criteria 67
   MID Server selection parameter 67
Netflow 30
netstat 30
network devices permissions 76
network devices rights 76
new algorithm 67
OId 183
owner comments 263
owner request 263
Parsing 1297
Pattern Designer 1311
plan 375
planned business service 383, 385, 387
planned entry points 383
Planner 375
planning
   create business service 381, 385
   discover business service 385
   fine-tune business service 387
   fix business service 387
   implement 387
   map business service 385
   planning business service 190, 195, 376, 377, 378, 380, 380
   planning phase 376, 377, 380

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
populate phase manually 380
potential business services 190, 195, 378, 380
PowerShell commands 374
Prerequisites 62
process 25
rctrlx.exe 67
Readiness 188
Readiness checklist 188
resolve errors 240
service 247
setup 367, 374
single application service 203
SNMP 183
subscription 64
supported applications 34, 397
track configuration files 751
traffic 343
traffic connections 30, 284
traffic related rule 30
traffic-based discovery 357, 359, 363
variables 1228
view configuration files 321, 327, 751
VPC logs 30
Service Mapping configuration 67
Service Mapping credentials 76
Service Mapping configuration
MID Server 67, 67, 72, 374, 374
service principal
Azure 1959
service status 1589
ServiceNow Discovery
reclassifying
workstation 560
workstation
reclassifying 560
ServiceNow Event Management manual service 1655
ServiceNow store 1442
services and dashboard 1370
ServiceWatch
activating 64
mapping flow 25
set up 1392
Set variable operation 1282
severity impact 1589
shared library 1106
Shazzam probe
configure 654
Show Similarities Details 1620
similar alerts 1619
similarity 1619
similarity solution 1621, 1623, 1626, 1627
Simple Notification Service 1544
simplify maps 335
single application service
creating 197, 203
mapping 197
skip 247
skip and resume 247
smart alert 1610, 1611, 1696, 1697, 1707
smart priority 1610, 1611, 1707
SNMP 183
SNMP probe
loading a MIB module 622
modules 623
parameters 620
SNMP query operation 1287
SNMP trap listener 1552
SNMP v1 traps 1552
SNMP-based queries 183
SNS 1544
Software Asset Management
configuring CIs to display SAM data 886
table schema and Discovery 885
Solaris
managing large payloads for 968
SolarWinds 1493
SolarWinds monitor 1491, 1493
SolarWinds Service & Application Monitor (SAM) 1491
solution 1024, 1356, 1625
spanning tree view 335
Split business service 269
SSH Command Probe 625, 625, 630, 630
Start self-health monitoring 1655
statement 608
status 1652
still receive events and generate alerts 1655
Stop self-health monitoring 1655
storage
data collected via host 939
Linux configuration for hosts of 938
Windows configuration for hosts of 938
storage path
Service Mapping 332
storage pattern, GCP 1083
Storage Pillar pattern, GCP 1083
storage volume resource table 2513
string 1214
subflow 1555, 1557, 1566
subflows 1562
subsequent alert 1596
sudo commands 118
super user 118
supported version 1442
system alert 1596
system properties
Cloud Insights 2549
T
tables 1359
Tables installed with Event Management 1359
tags
virtual resources 2307
task algorithm configuration 1677
task template 1570
task templates 1370
TBS 1369, 1376
technical service 1369
third-party applications 1442
third-party monitoring 1392
threshold 1657, 1659
ticket ID 1486
time of event 1394
time zone 1464
timeline 1707
timezone 1524
token 608
top down discovery credentials 76
prerequisites 76
top-down discovery credentials 76
prerequisites 76
top-down flow 25
topdown discovery credentials 76
prerequisites 76
topology 1598
topology map 1603, 1662
traffic connections 30, 284
traffic discovery 30, 284
traffic mapping 30, 284
traffic rule 30
traffic-based connection 1334
traffic-based noise 268
train 1621
training 1625
transform 1377, 1380, 1524
transform and compose 1413
transform event messages 1523
transform field values 1420
transform script 1545
transform table 1548
Transform table operation 1289
trigger recalculation 1610
troubleshoot map 247, 265
Troubleshooting 982
troubleshooting maps
Service Mapping 250
troubleshooting patterns
Service Mapping 250

U
unassociated events 1394
Unchange user operation 1291, 1291
unexclude 2548, 2548
unified log 1545
Union table operation 1292
Unix 1180, 1185
unknown type 1392
Update 1411
update set 1548
upgrade 1370, 1630, 1648
upgrade safe 1358
upload file 1284
URL format 1442, 1541
URL to push custom payloads 1524
use cURL to send events 1540
use JSON objects as data input and output 1526, 1541
user 1737
user access
  event management 286, 287, 1382, 1384
  service mapping 286, 287, 1382, 1384

V
V2 API commands 1185
value report 1640, 1734
vCenter
  probe upgrades 634
vCenter ESX Servers event rule 1494
Veritas Volume Manager
data discovered for 946
Vertical file parsing strategy 1299
video 1386
view map 1662, 1662
view service 1603
Virtual Machine 1544
Virtual Private Cloud 363
Virtual Private Cloud Flow Logs 363
virtual resources
tagging 2307
virtual server resource table 2513
Visibility 22
visualize 1630
VM 1544
VM virtual machine 1083, 1106
VMware
  ESX resource pools
    configuration 801
    relationships to without ESXi server 804
  virtual machines 804
VMware vCenter
discovery of 782
VMware vCenter server 1494
VPC Flow Logs 363
vRealize 1497

W
Web Performance Monitor (WPM) 1491
webhook 1547
WebLogic
  sudo 909
WebLogic servers
  Windows probes and sensors 910
  white list whitelist 1413
widgets 1630
Windows
data
  help the help desk 811
Windows operating system
  fetch data 647
WMI method invocation
  Pattern Designer 250, 276, 276, 1040, 1214, 1214,
    1214, 1214, 1219, 1219, 1219, 1219, 1227, 1227, 1227,
    1230, 1231, 1236, 1237, 1241, 1243, 1247, 1247, 1251,
    1255, 1257, 1261, 1277, 1282, 1284, 1287, 1289, 1291,
    1293, 1295, 1297, 1297, 1297, 1297, 1299, 1299, 1299,
    1301, 1301, 1301, 1303, 1303, 1303, 1303, 1305, 1305,
    1305, 1305, 1308, 1308, 1309, 1311, 1329, 1332,
    1342, 1342, 1342, 1343, 1343, 1343
  WMI method invocation operation 1293
  WMI query operation 1295
  word collection 1621
  word corpus 1621, 1623

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
work notes 1596, 1620, 1623, 1626, 1627
workflow 1557
workspace 1696, 1698
Workspace 1677, 1707, 1710
workstation
  reclassifying
    ServiceNow Discovery 560
    ServiceNow Discovery reclassifying 560

X
xml stats 1665
XMLStats 1665
xmlStats.do?include=XMLStatsEm 1665, 1666, 1667
XMLStatsEm 1666, 1667

Z
Zabbix 1499
Zabbix server 1498, 1499
zoom 1707