



Integration Service: ServiceNow SyncPacks

Version 3.0.0

Table of Contents

Introduction to ServiceNow SyncPacks	9
Architecture Overview for ServiceNow SyncPacks	10
SL1 and ServiceNow Terminology	11
Dependency Map for ServiceNow SyncPacks	11
Prerequisites for ServiceNow SyncPacks	12
Allowing Cross-Scoped Access	12
ServiceNow Base SyncPack	15
Downloading, Importing, and Installing the SyncPack	16
Downloading the SyncPack	16
Importing the SyncPack	16
Installing the SyncPack	17
Contents of the ServiceNow Base SyncPack	17
Integration Application	17
Configuration Object	17
Steps	17
ServiceNow Incident SyncPack	18
Downloading, Importing, and Installing the SyncPack	20
Downloading the SyncPack	20
Importing the SyncPack	20
Installing the SyncPack	21
Integration Applications Included in the SyncPack	21
Integration Applications	21
Integration Application (Internal)	22
Installing and Configuring Additional Components Used by the SyncPack	22
Installing the ScienceLogic SL1: CMDB & Incident Automation Application in ServiceNow	22
Installing the ServiceNow Base Pack PowerPack in SL1	23
Validating Network Communications	24
Checking DNS	24
Checking HTTPS and JSON	25
HTTP Codes	25
Creating a ServiceNow Group	26
Creating a ServiceNow User	28
Event Data Flow Between Integration Service and ServiceNow	30
Event Data Flow: Integration Service to ServiceNow	30
Event Data Flow: SL1 to Integration Service	32
Overview of the Run Book Automation for Incident Sync	33
Configuring Incident Sync	35
Creating and Aligning a Configuration Object	35
Creating a Configuration Object	36
Aligning a Configuration Object	39
Creating a ServiceNow Credential in SL1	41
Enabling the Run Book Automation Policies	43
Enabling and Customizing the Run Book Action Policy	44
Customizing the Snippet Code in the Input Parameters Pane	45
Customizing Logging in the Run Book Action	48
Sending Custom Data to ServiceNow Using the Passthrough Option	48
Passing Custom Data to ServiceNow	49
Snippet Code Example	51
Configuring the "ServiceNow: Click to Create Incident" Automation Policy	52
Viewing Incidents and Events	54

Incident Topology Suppression	55
Hyperlinking Events	61
ServiceNow Hyperlinking	62
Viewing the Incident Import Table in ServiceNow	62
ServiceNow, ScienceLogic Event, and Incident Priority Matrix	64
Adding Additional Fields to the Transform Map	65
ServiceNow CMDB SyncPack	71
Downloading, Importing, and Installing the SyncPack	72
Downloading the SyncPack	72
Importing the SyncPack	73
Installing the SyncPack	73
Integration Applications Included in the SyncPack	74
Integration Applications	74
Integration Applications (Internal)	75
Installing Additional Components Used by the SyncPack	76
Installing the ScienceLogic SL1: CMDB & Incident Automation Application in ServiceNow	76
Installing and Activating the CMDB Plugin in ServiceNow	77
Enabling the ServiceNow Identification and Reconciliation Module	78
Configuring Service Rules for Device Sync	78
Containment Rules	79
Hosting Rules	80
Creating a ServiceNow Update Set	81
Adding Service Rules to an Update Set	83
Exporting an Update Set	84
Installing the ScienceLogic Domain Separation (Global) Update Set in ServiceNow	85
Overview of the Update Set	86
Limitations of the Identification Engine	86
Installing the Update Set	87
Using ServiceNow Domain Separation with the Integration Service	87
User Setup	88
Example 1	88
Example 2	89
Workflow	90
Syncing Organizations	91
Syncing Organizations from SL1 to ServiceNow	91
Syncing Devices from SL1 to ServiceNow	93
Common Fields Used by Device Sync	93
Running a Device Sync	94
Adding Device Mappings with Postman	97
Persistently Saving Device Mappings with the API	99
Default Device Attribute Mappings	100
Syncing Custom Device Attributes	102
Adding New Device Attributes to ServiceNow	104
Syncing CI Attributes from ServiceNow to SL1	105
Syncing Advanced Topology Data from SL1 to ServiceNow	107
Syncing Network Interfaces from SL1 to ServiceNow	109
Syncing File Systems from SL1 to ServiceNow	111
Syncing Business Services from SL1 to ServiceNow	113
Syncing Installed Software between SL1 and ServiceNow	115
Discovery Sync	116
Configuring a ServiceNow Service Request for Discovery Sync	117
Discovery Sync Workflow	118

Running a Discovery Sync in ServiceNow	120
Discovering One or More Devices from ServiceNow to SL1	125
Decommissioning Devices	127
Activating the ServiceNow Service Request for Monitoring Removal	128
Removing Devices from Monitoring	128
Deleting Devices	130
Scheduling Integration Applications	131
Troubleshooting CMDB Sync	134
Issues Creating CIs in ServiceNow	134
Enabling Debugging of the Configuration Item Payload	136
Enabling Logs in ServiceNow	137
Log Messages for the "Generate Required CI Relations for ServiceNow" Application Integration	137
ServiceNow Change Management SyncPack	140
Downloading, Importing, and Installing the SyncPack	141
Downloading the SyncPack	141
Importing the SyncPack	141
Installing the SyncPack	142
Integration Applications Included in the SyncPack	142
Integration Applications	142
Integration Applications (Internal)	143
Syncing Device Maintenance from ServiceNow to SL1	143
Scheduling Device Maintenance	143
Directly Enabling or Disabling Device Maintenance from ServiceNow	148
Checklists for Deployment	150
CMDB-Only ServiceNow Integration with Single SL1 , no Domain Separation in ServiceNow	151
Sync Devices from SL1 to ServiceNow	151
Discover Devices from ServiceNow in SL1	152
Sync Business Services from SL1 to ServiceNow	152
Sync File Systems from SL1 to ServiceNow	152
Sync Network Interfaces from SL1 to ServiceNow	152
Sync Installed Software from SL1 to ServiceNow	153
Sync Maintenance Schedules from ServiceNow to SL1	153
Sync Advanced Topology from SL1 to ServiceNow	153
CMDB-Only ServiceNow Integration with Single SL1 and Domain-Separated ServiceNow	154
Sync Devices from SL1 to ServiceNow	154
Discover Devices from ServiceNow in SL1	155
Sync Business Services from SL1 to ServiceNow	155
Sync File Systems from SL1 to ServiceNow	156
Sync Network Interfaces from SL1 to ServiceNow	156
Sync Installed Software from SL1 to ServiceNow	156
Sync Maintenance Schedules from ServiceNow to SL1	156
Sync Advanced Topology from SL1 to ServiceNow	157
CMDB-Only ServiceNow Integration with Multiple SL1 Systems, no Domain-Separated ServiceNow	158
Sync Devices from SL1 to ServiceNow	159
Discover Devices from ServiceNow in SL1	159
Sync Business Services from SL1 to ServiceNow	160
Sync File Systems from SL1 to ServiceNow	160
Sync Network Interfaces from SL1 to ServiceNow	160
Sync Installed Software from SL1 to ServiceNow	161
Sync Maintenance Schedules from ServiceNow to SL1	161
Sync Advanced Topology from SL1 to ServiceNow	161
Incident-Only ServiceNow Integration with Single SL1 , no Domain Separation in ServiceNow	162

Certified Application Objects	163
Roles	164
Tables	165
Table Columns (cmdb_ci)	165
Table Columns (core_company)	166
Table Columns (cmdb_group)	166
Script Includes	166
Event Registry	166
Scripted Actions	167
Data Lookup Definitions	167
System Properties	167
Catalog Item	168
Catalog UI Policies	168
Variable Sets	168
Catalog Client Scripts	169
Workflows	169
Scripted REST Resources	170
Transform Maps	172
Transform Scripts	172
ServiceNow API Endpoints	173
Business Services	175
HTTP Method	175
Pagination	175
Resource Path	175
Default Resource Path	175
Example (Request URL)	175
Example (Response)	176
Change Requests	177
HTTP Method	177
Resource Path	177
Default Resource Path	177
Fixed Internal Query	178
Example	178
Example (Response)	178
Classification version 1	179
HTTP Method	179
Pagination	179
Resource Path	179
Default Resource Path	179
Fixed Internal Query	180
Example	180
Example (Response)	180
Classification version 2	182
HTTP Method	182
Pagination	182
Resource Path	182
Default Resource Path	182
Fixed Internal Query	183
Example	183
Example (Response)	183
CMDB Group	185
HTTP Method	185

Pagination	185
Resource Path	185
Default Resource Path	185
Example (Request URL)	185
Example (Body)	185
Example (Response)	186
Companies	187
HTTP Method	187
Pagination	187
Resource Path	187
Default Resource Path	187
Fixed Internal Query	188
Example	188
Example (Response)	188
Configuration Items	189
HTTP Method	189
Pagination	189
Resource Path	189
Default Resource Path	189
Fixed Internal Query	190
Example	190
Example (Response)	190
Device Identification Engine	191
HTTP Method	191
Pagination	191
Resource Path	191
Default Resource Path	191
Example (Request URL)	191
Example (Body)	191
Example Business Service (Body)	192
Example (Response)	193
File Systems	194
HTTP Method	194
Pagination	194
Resource Path	194
Default Resource Path	194
Fixed Internal Query	195
Example	195
Example (Response)	195
Import Set	196
HTTP Method	196
Resource Path	196
Default Resource Path	196
Example (Request URL)	196
Example (Body)	196
Incidents	198
HTTP Method	198
Pagination	198
Resource Path	198
Default Resource Path	198
Fixed Internal Query	199
Example	199

Example (Response)	199
Installed Software	200
HTTP Method	200
Pagination	200
Resource Path	200
Default Resource Path	200
Fixed Internal Query	201
Example (Request URL)	201
Example (Response)	201
Manufacturer	203
HTTP Method	203
Pagination	203
Resource Path	203
Default Resource Path	203
Example (Request URL)	203
Example (Body)	203
Example (Response)	204
Model	205
HTTP Method	205
Resource Path	205
Default Resource Path	205
Example (Request URL)	205
Example (Body)	205
Example (Response)	206
Network Adapters	207
HTTP Method	207
Pagination	207
Resource Path	207
Default Resource Path	207
Fixed Internal Query	208
Example (Request URL)	208
Example (Response)	208
Service Requests	209
HTTP Method	209
Pagination	209
Resource Path	209
Default Resource Path	209
Fixed Internal Query	210
Example	210
Example (Response)	210
ServiceNow Registered Events	212
Catalog Item Events	213
x_sclo_scilogic.device_monitoring	213
Trigger	213
Command	213
Event Fields	213
Example	213
x_sclo_scilogic.remove_monitoring	214
Trigger	214
Command	214
Event Fields	214
Example	214

Maintenance Mode Events	215
x_sclo_scilogic.device_maintenance	215
Trigger	215
Command	215
Event Fields	215
Example	215

Chapter

1

Introduction to ServiceNow SyncPacks

Overview

This manual describes the *Integration Service: ServiceNow SyncPacks*. Previously, these SyncPacks were contained in the *ServiceNow SyncPack*, but with the release of Integration Service version 2.0.0, these SyncPacks were separated into their own self-contained packages to allow for easier uploading, installing, and upgrading.

The available SyncPacks for ServiceNow integration include:

- *ServiceNow Base SyncPack*
- *ServiceNow Incident SyncPack*
- *ServiceNow Configuration Management Database (CMDB) SyncPack*
- *ServiceNow Change Management SyncPack*

To locate the available ServiceNow SyncPacks, go to the ScienceLogic Support site at <https://support.sciencelogic.com/s/>. Click the **Product Downloads** tab, select **PowerPacks**, and then click the *Synchronization* link.

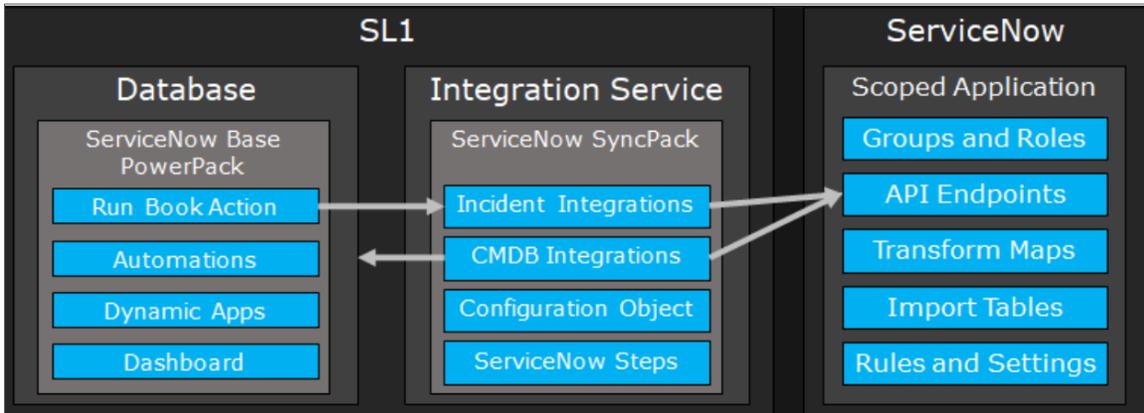
NOTE: For checklists that cover installation and deployment steps for specific configurations of the Integration Service, ServiceNow, and SL1, see [Appendix A: Checklists for Deployment](#).

This chapter covers the following topics:

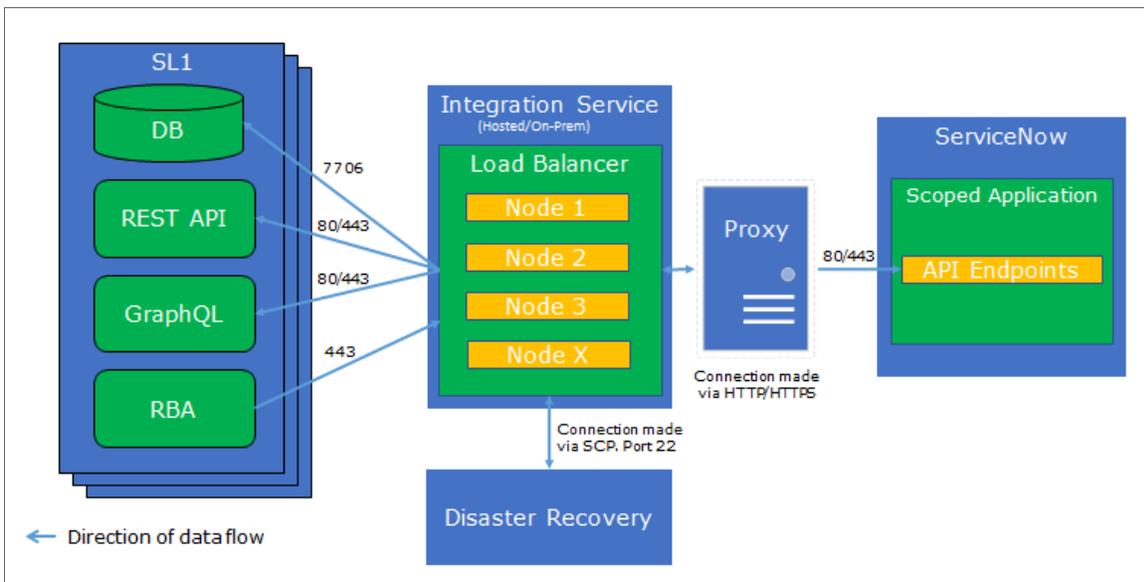
Architecture Overview for ServiceNow SyncPacks	10
Prerequisites for ServiceNow SyncPacks	12
Allowing Cross-Scoped Access	12

Architecture Overview for ServiceNow SyncPacks

The following diagram details the various elements that are contained in SL1 and the Integration Service, and how the Integration Service sits between the core SL1 platform and an external data platform:



The following diagram provides an example of the high-level architecture of an Integration Service system with High Availability, Disaster Recovery, and a proxy configured:



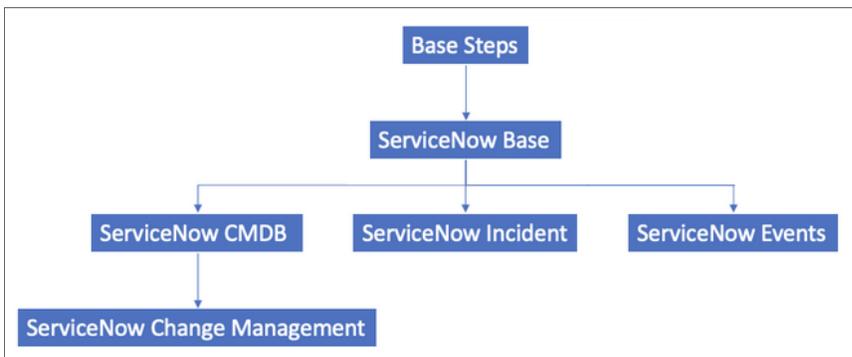
SL1 and ServiceNow Terminology

The following table lists the different names for the shared elements in SL1 and ServiceNow:

SL1	ServiceNow
Event	Incident
Device	CI (Configuration Item)
Asset	Asset
Organization	Company/Domain
Discovery Session	Service Catalog Request
Schedule	Maintenance Schedule
Topology	Dependency

Dependency Map for ServiceNow SyncPacks

The following graphic describes which SyncPacks require you to install one or more SyncPacks first before using those SyncPacks:



TIP: For more information about the *Standard Base Steps SyncPack*, see the *Integration Service Platform* manual.

Prerequisites for ServiceNow SyncPacks

This section describes the prerequisites for the ServiceNow SyncPacks. For more information about the specific software versions required by a ServiceNow SyncPack, see the chapter for that SyncPack in this manual.

To install the ScienceLogic ServiceNow SyncPacks, you must have administrator access to both SL1 and ServiceNow. Specifically, you will need:

- ScienceLogic root SSH access
- SSH access to the Integration Service
- ScienceLogic administrator access to the Administration Portal
- ServiceNow administrator access

Additional information related to uploading and installing the ServiceNow SyncPacks:

- If you want to upload and install multiple ServiceNow SyncPacks at the same time, you should upload *all* of the SyncPacks first, and then install them to address any dependencies between the SyncPacks.
- If you are upgrading from version 1.8.4 of the Integration Service, you can first move to version 2.5.0 of the *ServiceNow SyncPack*, then upgrade to version 2.0 of the Integration Service, and finally move to version 3.0.0 of the relevant ServiceNow SyncPacks.
- If you made customizations to version 2.5.0 or earlier of the *ServiceNow SyncPack*, you must make a copy of that SyncPack and save it as a new SyncPack to keep your customizations. For more information, see the ***Integration Service for Developers*** manual.
- If you are starting out with version 2.0.0 of the Integration Service platform, you should skip version 2.5.0 of the *ServiceNow SyncPack* and instead install version 3.0.0 of the relevant ServiceNow SyncPacks.
- There is no backwards compatibility after you upgrade a ServiceNow SyncPack from version 2.5.0 to 3.0.0.

<p>NOTE: ScienceLogic highly recommends that you disable all firewall session-limiting policies. Firewalls will drop HTTPS requests, which results in data loss.</p>

Allowing Cross-Scoped Access

When using custom tables that are extended from the `cmdb_ci` table, you must configure cross-scope access for any custom tables created outside of the base ServiceNow deployment.

The following examples contain errors that might occur when cross-scope access is required.

Example of an API response:

```
{"results":[{"error": {"message": "com.glide.script.fencing.access.ScopeAccessNotGrantedException: read access to ui_test_hardware not granted", "detail": ""}, "status": "failure"}
```

Example of navigating to a URL directly from a web browser when cross-scope access is required:

This page contains the following errors:

error on line 1 at column 1: Document is empty

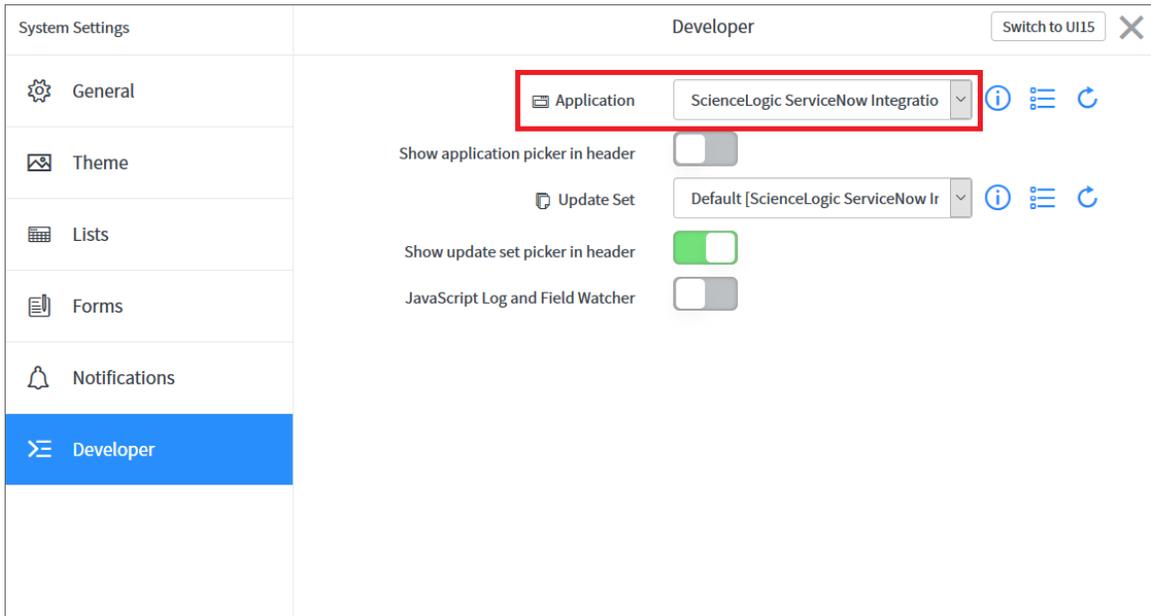
Below is a rendering of the page up to the first error.

In this example, the table requires that you grant access to the ScienceLogic Scope to allow the API call to run correctly. In the above example, the target table is **u_test_hardware**.

NOTE: A ServiceNow account with System Administrator is required.

To grant access to the ScienceLogic Scope in ServiceNow:

1. Log in to your ServiceNow instance.
2. Click the **Settings** icon (⚙️) and select the **Developer** tab. The **Developer System Settings** window appears:



3. From the **Application** drop-down list, select *ScienceLogic ServiceNow Integration*.
4. Close the **Developer System Settings** window and navigate to the **Cross scope privileges** page (System Applications > Application Cross-Scope Access).

NOTE: Make sure you are in the ScienceLogic ServiceNow Application scope and track these updates in an update set.

5. Click the **[New]** button to create a new record on the **Cross scope privileges** page:

The screenshot shows the 'Cross scope privilege' form in ServiceNow. The form is titled 'Cross scope privilege' and 'New record'. It contains several fields: 'Source Scope' (ScienceLogic ServiceNow Integration), 'Application' (ScienceLogic ServiceNow Integration), 'Target Scope' (Global), 'Operation' (Read), 'Target Name' (cmd_ci_endpoint_storage_fc), and 'Status' (Allowed). The 'Target Type' is set to 'Table'. A 'Submit' button is visible at the bottom left.

6. Verify that the **Source Scope** and **Application** fields are set to *ScienceLogic ServiceNow Integration*. If they are not, repeats steps 2-3.
7. Complete the following fields:
 - **Target Scope**. Specify the scope of the target table, such as *Global*.
 - **Operation**. Select *Read*.
 - **Target Name**. Specify the name of the target table.
 - **Status**. Select *Allowed*.
 - **Target Type**. Select *Table*.
8. Click the **[Submit]** button.

For more information, see https://docs.servicenow.com/bundle/madrid-application-development/page/build/applications/reference/c_CrossScopePrivilegeRecord.html.

Chapter

2

ServiceNow Base SyncPack

Overview

This chapter describes the *ServiceNow Base SyncPack*, which contains a set of integration applications, steps, a configuration object, and other utilities that are common to the entire ServiceNow Integration.

You must install the *ServiceNow Base SyncPack* before you can install any of the other ServiceNow SyncPacks.

This chapter covers the following topics:

<i>Downloading, Importing, and Installing the SyncPack</i>	16
<i>Contents of the ServiceNow Base SyncPack</i>	17

Downloading, Importing, and Installing the SyncPack

A SyncPack file has the **.whl** file extension type. You can download the SyncPack file from the ScienceLogic Support site.

Downloading the SyncPack

To locate and download the SyncPack:

1. Go to the ScienceLogic Support site at <https://support.sciencelogic.com/s/>.
2. Click the **Product Downloads** tab, select *PowerPacks*, and then click the "Synchronization" link. The **Synchronization PowerPack Downloads** page appears.
3. Select the SyncPack you want to install.
4. In the **Files** list, select the SyncPack **.whl** file and click **[Download]**.

NOTE: SyncPacks do not require a specific license. After you download a SyncPack, you can import it to your Integration Service using the Integration Service user interface.

Importing the SyncPack

NOTE: You must import and install the *ServiceNow Base SyncPack* before uploading and installing any of the other ServiceNow SyncPacks.

To import a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click **[Import SyncPack]**. The **Import SyncPack** page appears.
2. Click **[Browse]** and select the **.whl** file for the SyncPack you want to install.

TIP: You can also drag and drop a **.whl** file to the **SyncPacks** page.

3. Click **[Import]**. The Integration Service registers and uploads the SyncPack. The SyncPack is added to the **SyncPacks** page.

NOTE: You cannot edit the content package in a SyncPack published by ScienceLogic. You must make a copy of a ScienceLogic SyncPack and save your changes to the new SyncPack to prevent overwriting any information in the original SyncPack when upgrading.

Installing the SyncPack

To install a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click the **[Actions]** button () for the SyncPack you want to install and select *Activate & Install*. The **Activate & Install SyncPack** modal appears.

TIP: By default, the **SyncPacks** page only displays activated and installed SyncPacks. If you do not see the SyncPack that you want to install, click the toggle icon () on the **SyncPacks** page and select *Show All SyncPacks* to see a list of the uninstalled SyncPacks.

2. Click **[Yes]** to confirm the activation and installation. When the SyncPack is activated, the **SyncPacks** page displays a green check mark icon () for that SyncPack. If the activation or installation failed, then a red exclamation mark icon () appears.
3. For more information about the activation and installation process, click the check mark icon () or the exclamation mark icon () in the **Activated** column for that SyncPack. For a successful installation, the "Activate & Install SyncPack" integration application appears, and you can view the Step Log for the steps. For a failed installation, the **Error Logs** window appears.

Contents of the ServiceNow Base SyncPack

This section lists the contents of the *ServiceNow Base SyncPack*.

Integration Application

- **Cache SL1 Users.** Performs a query for all existing Integration Service users and writes them to a cache.

Configuration Object

- **ServiceNow SyncPack.** Contains a set of sample configuration data, including the required block of SL1 hostname code needed for SL1 8.12.0 or later.

Steps

- Merge and Chunk Payloads for ServiceNow
- MySQL Multiple
- Optional QueryGQL Call
- Process IS Configs
- Query and Cache SL1 User Info

Chapter

3

ServiceNow Incident SyncPack

Overview

This chapter describes the *ServiceNow Incident SyncPack*, which is the ScienceLogic integration with the ServiceNow Incident Management Module.

This SyncPack automatically logs, de-duplicates, correlates, updates, and appends ServiceNow Incidents, reducing the amount of time to resolve critical service issues. This SyncPack covers the entire Incident life cycle, providing a bi-directional integration between SL1 events and ServiceNow Incidents, while providing a granular view into both the event and the associated Incident.

This chapter covers the following topics:

<i>Downloading, Importing, and Installing the SyncPack</i>	20
<i>Integration Applications Included in the SyncPack</i>	21
<i>Installing and Configuring Additional Components Used by the SyncPack</i>	22
<i>Event Data Flow Between Integration Service and ServiceNow</i>	30
<i>Overview of the Run Book Automation for Incident Sync</i>	33
<i>Configuring Incident Sync</i>	35
<i>Creating and Aligning a Configuration Object</i>	35
<i>Creating a ServiceNow Credential in SL1</i>	41
<i>Enabling the Run Book Automation Policies</i>	43
<i>Enabling and Customizing the Run Book Action Policy</i>	44
<i>Sending Custom Data to ServiceNow Using the Passthrough Option</i>	48
<i>Configuring the "ServiceNow: Click to Create Incident" Automation Policy</i>	52
<i>Viewing Incidents and Events</i>	54

<i>Incident Topology Suppression</i>	55
<i>Hyperlinking Events</i>	61
<i>Viewing the Incident Import Table in ServiceNow</i>	62
<i>ServiceNow, ScienceLogic Event, and Incident Priority Matrix</i>	64
<i>Adding Additional Fields to the Transform Map</i>	65

Downloading, Importing, and Installing the SyncPack

A SyncPack file has the **.whl** file extension type. You can download the SyncPack file from the ScienceLogic Support site.

Downloading the SyncPack

To locate and download the SyncPack:

1. Go to the ScienceLogic Support site at <https://support.sciencelogic.com/s/>.
2. Click the **Product Downloads** tab, select *PowerPacks*, and then click the "Synchronization" link. The **Synchronization PowerPack Downloads** page appears.
3. Select the SyncPack you want to install.
4. In the **Files** list, select the SyncPack **.whl** file and click **[Download]**.

NOTE: SyncPacks do not require a specific license. After you download a SyncPack, you can import it to your Integration Service using the Integration Service user interface.

Importing the SyncPack

NOTE: You must import and install the *ServiceNow Base SyncPack* before uploading and installing any of the other ServiceNow SyncPacks.

To import a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click **[Import SyncPack]**. The **Import SyncPack** page appears.
2. Click **[Browse]** and select the **.whl** file for the SyncPack you want to install.

TIP: You can also drag and drop a **.whl** file to the **SyncPacks** page.

3. Click **[Import]**. The Integration Service registers and uploads the SyncPack. The SyncPack is added to the **SyncPacks** page.

NOTE: You cannot edit the content package in a SyncPack published by ScienceLogic. You must make a copy of a ScienceLogic SyncPack and save your changes to the new SyncPack to prevent overwriting any information in the original SyncPack when upgrading.

Installing the SyncPack

To install a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click the **[Actions]** button () for the SyncPack you want to install and select *Activate & Install*. The **Activate & Install SyncPack** modal appears.

TIP: By default, the **SyncPacks** page only displays activated and installed SyncPacks. If you do not see the SyncPack that you want to install, click the toggle icon () on the **SyncPacks** page and select *Show All SyncPacks* to see a list of the uninstalled SyncPacks.

2. Click **[Yes]** to confirm the activation and installation. When the SyncPack is activated, the **SyncPacks** page displays a green check mark icon () for that SyncPack. If the activation or installation failed, then a red exclamation mark icon () appears.
3. For more information about the activation and installation process, click the check mark icon () or the exclamation mark icon () in the **Activated** column for that SyncPack. For a successful installation, the "Activate & Install SyncPack" integration application appears, and you can view the Step Log for the steps. For a failed installation, the **Error Logs** window appears.

Integration Applications Included in the SyncPack

This section lists the contents of the *ServiceNow Incident SyncPack*.

Integration Applications

The following integration applications are included with the *ServiceNow Incident SyncPack*:

- **Create or Update ServiceNow Incident from SL1 Event.** Reads SL1 events and creates or updates the corresponding ServiceNow Incident.
- **Sync Incident State from ServiceNow to SL1 Event.** Clears or updates SL1 events when the related ServiceNow Incident is updated.
- **Update ServiceNow Incident when SL1 Event is Acknowledged.** Updates the synced ServiceNow incident when the corresponding SL1 event is acknowledged.
- **Update ServiceNow Incident when SL1 Event is Cleared.** Updates the synced ServiceNow Incident when the corresponding SL1 event is cleared.

For more information about these integration applications, see [Configuring Incident Sync](#).

Integration Application (Internal)

To view the internal integration application, click the Filter icon () on the **Integrations** page and select *Show Hidden Integrations*. Internal integration applications are hidden by default. The following integration application is "internal" and should not be run directly. Instead, it is automatically run by applications from the previous list:

- **Bulk Update SL1 Events**. Bulk updates SL1 events with a given payload.

Installing and Configuring Additional Components Used by the SyncPack

This workflow covers how to install and configure the different elements used with the ServiceNow Incident SyncPack:

1. [Install the ScienceLogic SL1: CMDB & Incident Automation application in ServiceNow](#)
2. [Install the ServiceNow Base Pack PowerPack in SL1](#)
3. [Validate network communications](#)
4. [Create a ServiceNow group and user account](#)

NOTE: For checklists that cover installation and deployment steps for specific configurations of the Integration Service, ServiceNow, and SL1, see [Appendix A: Checklists for Deployment](#).

Installing the ScienceLogic SL1: CMDB & Incident Automation Application in ServiceNow

The *ServiceNow Incident SyncPack* uses the "ScienceLogic SL1: CMDB & Incident Automation Application" to sync incident status update from ServiceNow back to SL1.

You must install the "ScienceLogic SL1: CMDB & Incident Automation" application on your ServiceNow instance to enable this SyncPack. The "ScienceLogic SL1: CMDB & Incident Automation" application is also known as the "Certified Application" or the "Scoped Application".

NOTE: You must have a ServiceNow HI Service Account to request this application and download it onto your ServiceNow instance.

WARNING: Integration Service instances running version 2.0.0 or later of the ServiceNow integration applications are not backwards-compatible with the previous ServiceNow update sets or with SyncServer. After you install the "ScienceLogic SL1: CMDB & Incident Automation" application on your ServiceNow instance, you need to upgrade your ServiceNow integration applications to version 2.0.0 or later on all Integration Service instances. The "ScienceLogic SL1: CMDB & Incident Automation" application is also not backwards-compatible with SyncServer. This change cannot be reverted.

You must first request the "ScienceLogic SL1: CMDB & Incident Automation" application from the ServiceNow Store, and then you can install it.

To request and install the Certified Application:

1. Go to the ServiceNow Store at <https://store.servicenow.com> and search for "ScienceLogic SL1".
2. Select the "ScienceLogic SL1: CMDB & Incident Automation" application. The detail page for the application appears.
3. Click the **[Get]** button and log in with your HI credentials.
4. After the request is approved, log in to ServiceNow as an administrator and navigate to **Application Manager** (System Applications > Applications).
5. Click **[Downloads]** in the menu header or search for "ScienceLogic".
6. Click the version drop-down for the "ScienceLogic ServiceNow Integration" application listing to make sure you are using the correct version of the application that is compatible with your version of this SyncPack.
7. Click the **[Install]** button for the "ScienceLogic ServiceNow Integration" application. The installation is complete when the button changes to **[Installed]**.
8. In the filter navigator, search for "ScienceLogic" and locate the application in the left-hand navigation menu.

Installing the ServiceNow Base Pack PowerPack in SL1

The *ServiceNow Base Pack PowerPack* monitors the ServiceNow Incident and CMDB tables, and it returns information about Incident types, priorities, and states, displaying the information in an easy-to-consume dashboard. The PowerPack also returns information about the CI records that are actively being synced between SL1 and ServiceNow via the Integration Service, including basic CI metadata as well as an overall count.

Also, this PowerPack is a critical component of the Incident Sync Integration with ServiceNow, using Run Book Automations to integrate with the ServiceNow Incident Management module.

TIP: By default, installing a new version of a PowerPack overwrites all content in that PowerPack that has already been installed on the target system. You can use the **Enable Selective PowerPack Field Protection** setting in the **Behavior Settings** page (System > Settings > Behavior) to prevent new PowerPacks from overwriting local changes for some commonly customized fields. For more information, see the **System Administration** manual.

To install the *ServiceNow Base Pack* PowerPack:

1. Download the latest version of the PowerPack from the Customer Portal to a local computer.
2. Log in to SL1 as an administrator, then go to the **PowerPack Manager** page (System > Manage > PowerPacks).
3. Click the **[Actions]** button and select *Import PowerPack*.
4. Click the **[Browse]** button and navigate to the *ServiceNow Base Pack* PowerPack file from step 1.
5. Select the PowerPack file and click **[Import]**. The **PowerPack Installer** modal page displays a list of the PowerPack contents.
6. Click the **[Install]** button. After the installation is complete, the *ServiceNow Base Pack* PowerPack appears on the **PowerPack Manager** page.

Validating Network Communications

All communication between SL1 and ServiceNow is done through TCP port 443. To allow communication between SL1 and ServiceNow, the SL1 Database Server, Data Collector, or All-In-One Appliance must have external access to the ServiceNow instance. No inbound TCP ports are required to be open to the SL1 server. Outbound communication may use NAT or be direct.

NOTE: All firewall session-limiting policies must be disabled. If firewall session-limiting policies are enabled, HTTPS requests might be dropped by the firewall, resulting in data loss. Check with your security or firewall administrator to make sure there are no session limiting policies on TCP port 443 for your SL1 servers.

Checking DNS

Because ServiceNow is a cloud-based service, DNS must be configured on all SL1 servers that communicate with your ServiceNow instance.

NOTE: ServiceNow instances are generally named as: ***your-instance.service-now.com***, where ***your-instance*** is the name of your ServiceNow server. The examples below use ***mycompany.service-now.com***. Your instance name will be unique to your subscription.

To validate that your SL1 server has proper DNS name resolution configured, test network connectivity and name resolution using the `nmap` command, which is available from the command line of any SL1 server:

```
nmap -sT -p 443 mycompany.service-now.com
```

If the test was successful, you will see a message similar to the following:

```
Starting Nmap 5.51 ( http://nmap.org ) at 2013-11-12 20:22 UTC
Nmap scan report for mycompany.service-now.com (199.91.136.100)
Host is up (0.067s latency).
PORT STATE SERVICE
443/tcp open  https
```

If domain name resolution fails, you will see a message similar to:

```
Failed to resolve given hostname/IP: mycompany.service-now.com.
```

Checking HTTPS and JSON

You can administer a simple test to determine if the ServiceNow JSON Plug-in web service is configured and operating using the Basic Authentication method on your ServiceNow instance. To do so, run the following command from the ScienceLogic Central Database or All-In-One Appliance:

NOTE: In the example below, replace the `admin:admin` username and password key/value pair with your ServiceNow administrator username and password and `mycompany.service-now.com` with your ServiceNow instance name.

```
curl --location -vu admin:admin -H "Accept: application/json" -H "Content-Type: application/json" 'https://mycompany.service-now.com/api/now/table/incident'
```

If not successful, the following message appears:

```
HTTP/1.1 401 Unauthorized
```

If successful, a JSON encoded string starting with the "result" variable appears:

```
{"result":[{"upon_approval":"","location":"1083361cc611227501b682158cabf646",...
```

HTTP Codes

HTTP codes are necessary for identifying specific problems. The following table lists typical HTTP codes that might occur when testing the ServiceNow JSON Web Service.

Code	Definition
401	Unauthorized. Check that the username and password are correct and properly formatted.
403	Forbidden. ServiceNow understood the request, but either the URL is incorrect, or the user account does not have permission to see the requested object.
404	The ServiceNow server has not found anything matching the requested URL. Check to make sure there is data in the target table.
200	Success.
201	Success. Data is posted.

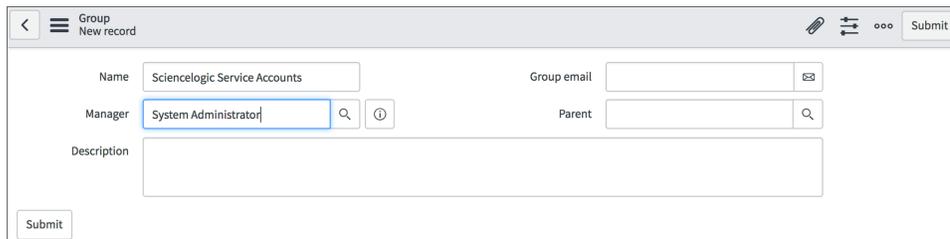
TIP: For more information about the ServiceNow JSON Web Service and the Table API, see http://wiki.servicenow.com/index.php?title=Table_API. If you continue to have problems, please contact either ScienceLogic or ServiceNow customer support.

Creating a ServiceNow Group

For best practice and security, create a dedicated ServiceNow account that has restricted access to only the groups, access control lists (ACLs), and roles needed for ScienceLogic incident management.

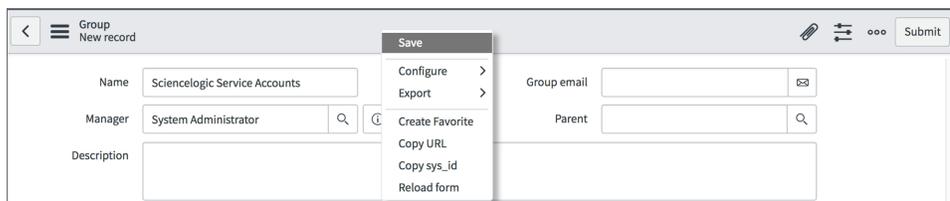
To create a ServiceNow Account for ScienceLogic Incident management:

1. In ServiceNow, search in the filter navigator for "groups".
2. On the **Groups** page (System Security > Groups), click **[New]**. A **New record** page appears.
3. In the **New record** page, type the group name and any additional information. **Name** is the only required field.



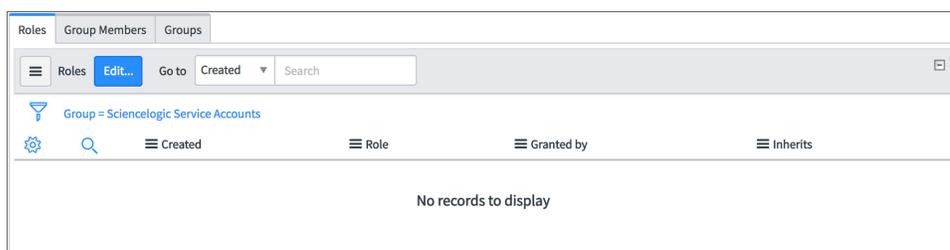
The screenshot shows the 'Group New record' form in ServiceNow. The 'Name' field is filled with 'Sciencelogic Service Accounts'. The 'Manager' field is filled with 'System Administrator'. The 'Parent' field is empty. There are search icons next to the 'Manager' and 'Parent' fields. A 'Submit' button is located at the bottom left of the form.

4. Right-click the gray header and click **Save** to save the record.



The screenshot shows the 'Group New record' form with a context menu open over the header. The 'Save' option is selected. Other options in the menu include 'Configure', 'Export', 'Create Favorite', 'Copy URL', 'Copy sys_id', and 'Reload form'.

5. At the bottom of the Group form, select the **[Roles]** tab and click **[Edit]**.



The screenshot shows the 'Roles' tab for the 'Group = Sciencelogic Service Accounts'. The 'Roles' tab is selected, and the 'Edit...' button is highlighted. The main content area shows 'No records to display'.

6. Search for `x_sclo_scilogic.Admin` and move it to the **Roles List** column using the arrow buttons.

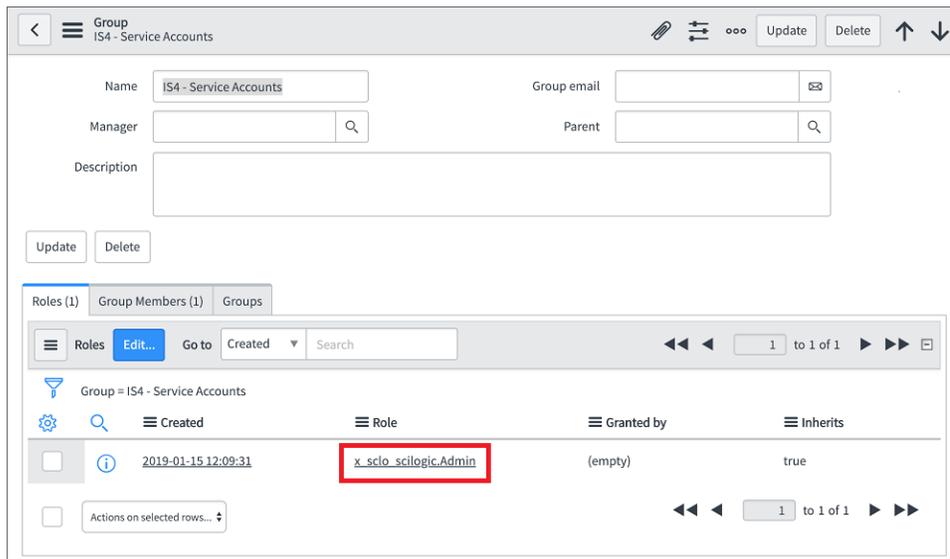
The screenshot shows a user interface for managing roles. At the top, there are two buttons: "Add Filter" and "Run filter" with a help icon. Below these are three dropdown menus: "-- choose field --", "-- oper --", and "-- value --".

The interface is divided into two main sections:

- Collection:** Contains a search bar with a magnifying glass icon. Below it is a list of roles: `action_designer`, `activity_admin`, `activity_creator`, `admin`, `agent_admin`, `announcement_admin`, `api_analytics_read`, `app_service_admin`, `app_service_user`, `approval_admin`, `approver_user`, `assessment_admin`, `asset`, `assignment_rule_admin`, `atf_test_admin`, and `atf_test_designer`.
- Roles List:** Contains the text "IS4 - Service Accounts" and a list with one role: `x_sclo_scilogic.Admin`, which is highlighted with a red border.

Between the two lists are two arrow buttons: a right-pointing arrow (>) and a left-pointing arrow (<). At the bottom of the interface are two buttons: "Cancel" and "Save".

7. Click **[Save]**. Your ServiceNow Group now has an assigned Role:



8. Next, create a ServiceNow user to use with this Group. See the following procedure for the details.

Creating a ServiceNow User

NOTE: The ServiceNow user you create in this procedure will *not* be able to log into the ServiceNow user interface with the username and password you give this user. However, you will use the username and password in the relevant configuration objects in the Integration Service user interface to run integration applications. For more information about configuration objects, see [Creating and Aligning a Configuration Object](#).

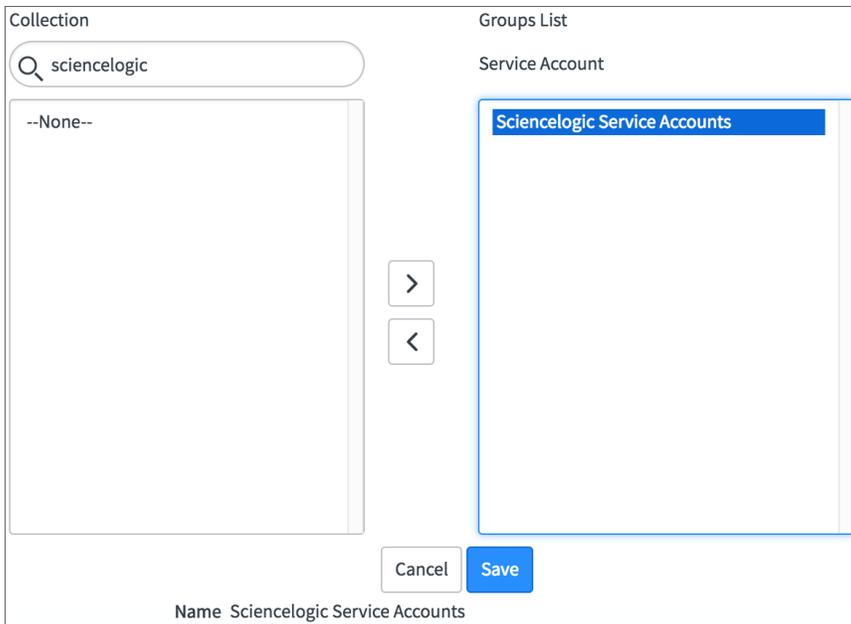
To create a ServiceNow Account for ScienceLogic Incident management:

1. In ServiceNow, search in the filter navigator for "users".
2. On the **Users** page (System Security > Users), click **[New]**. A **New record** page appears.
3. Complete the following fields:
 - **User ID**. Type a user ID. Required.
 - **First Name**. Type the user's first name.
 - **Last Name**. Type the user's last name.
 - **Password**. Type a password. Required.
 - **Active**. Select this checkbox. Required.
 - **Web Service Access Only**. Select this checkbox. Required.
 - **Time Zone**. Select *GMT*. Required.
 - **Date Format**. Select *System (yyyy-MM-dd)*.

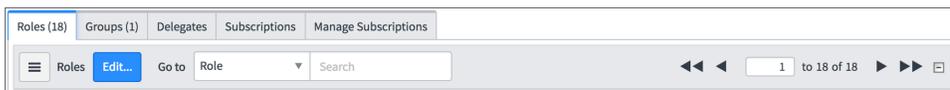
4. Right-click the gray header and click Save to save the user.
5. Select the **[Groups]** tab at the bottom of the record and click the **[Edit]** button:



6. Find the group you created previously and move the group to the right-hand column using the arrow buttons.



7. Click **[Save]**. After the user has been added to the group, you can see their Roles and Groups at bottom of the record:

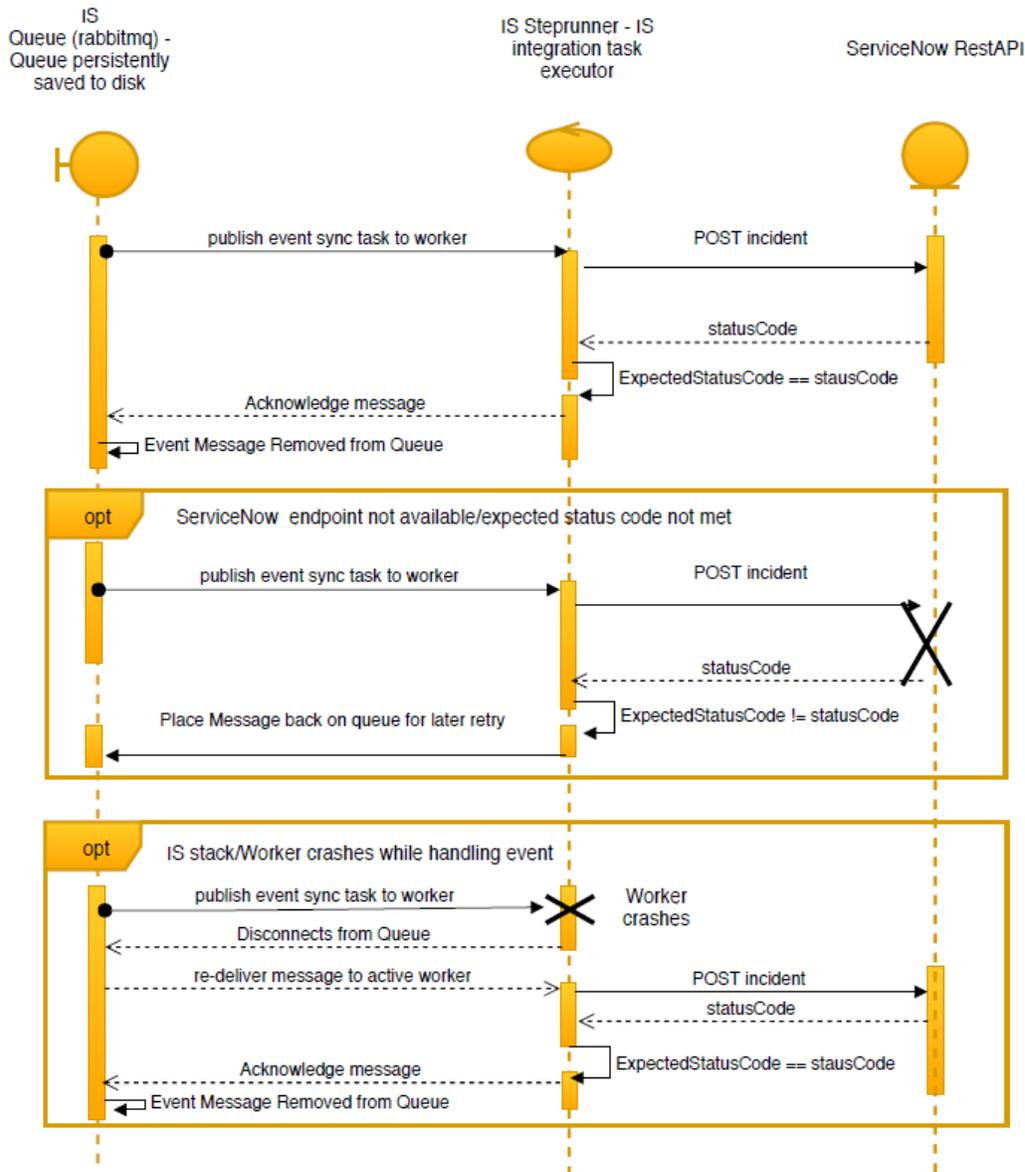


NOTE: As a best practice, you should use a non-administrator ServiceNow user for the Integration Service configuration object.

Event Data Flow Between Integration Service and ServiceNow

Event Data Flow: Integration Service to ServiceNow

The following chart and steps illustrate the event data flow from the Integration Service to a ServiceNow instance:



1. Workers are subscribed to the Integration Service task queue.
2. When a new event to be synced is placed in the Integration Service task queue, it is assigned and pushed to a worker.

3. The worker processes and transforms the necessary SL1 event data into a ServiceNow incident and POSTs the incident to the ServiceNow endpoint.
 - If the resulting status code matches the expected status code for the request, the original message is acknowledged and removed from the queue.
 - If the worker crashes while processing the event, the queue senses the unexpected disconnect, and the same event message is re-delivered to a new worker.

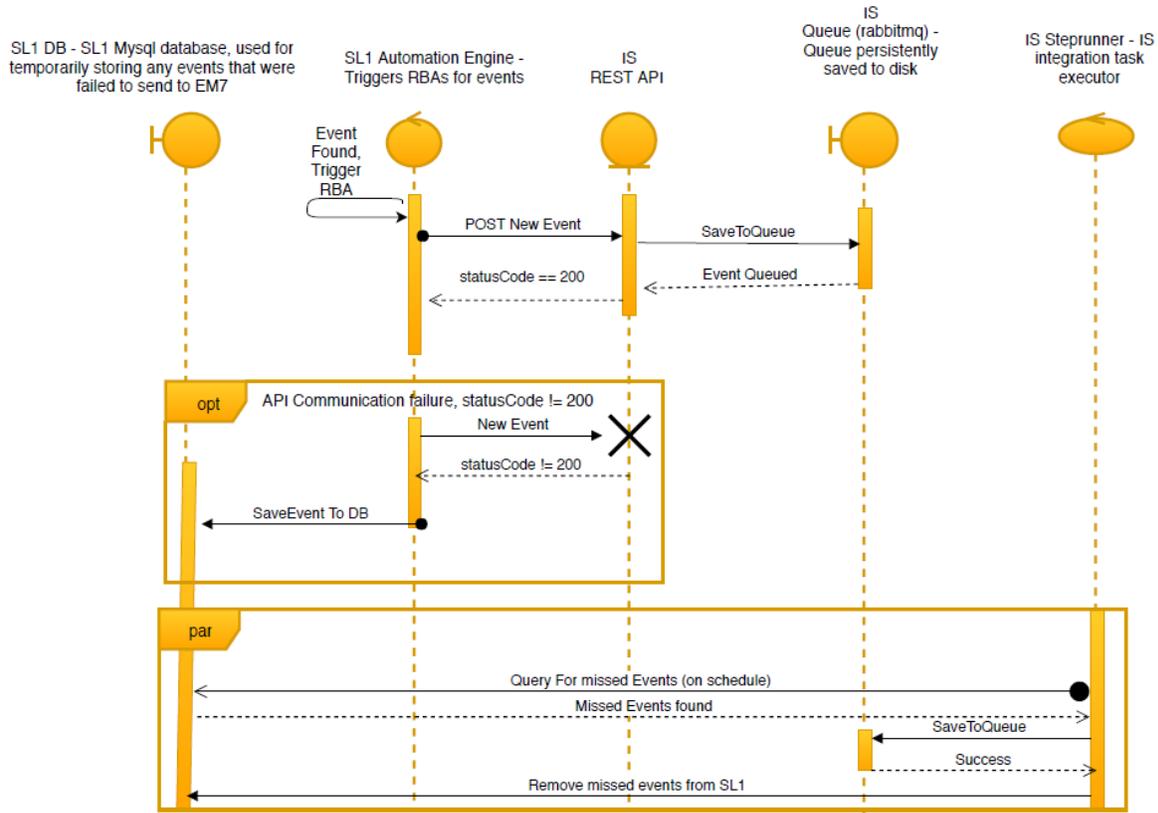
NOTE: The above results are performed through the "late acknowledgment" of tasks. With this setting enabled, an Integration Service worker will not remove a message from the queue until the message has been fully processed by the worker. This setting can be enabled or disabled with the environment variable "task_acks_late".

4. If ServiceNow responds with an unexpected status code when POSTing the incident, the message will be placed back in the queue with specified re-try parameters.

NOTE: You can configure re-try parameters on a per-task basis. You may want to manually alter your re-try parameters for tasks depending on the action the task is taking. The configuration of retries includes the maximum number of times a task is retried after consistently failing, and the delay length between retries.

Event Data Flow: SL1 to Integration Service

The following chart and steps illustrate the event data flow from SL1 to the Integration Service:



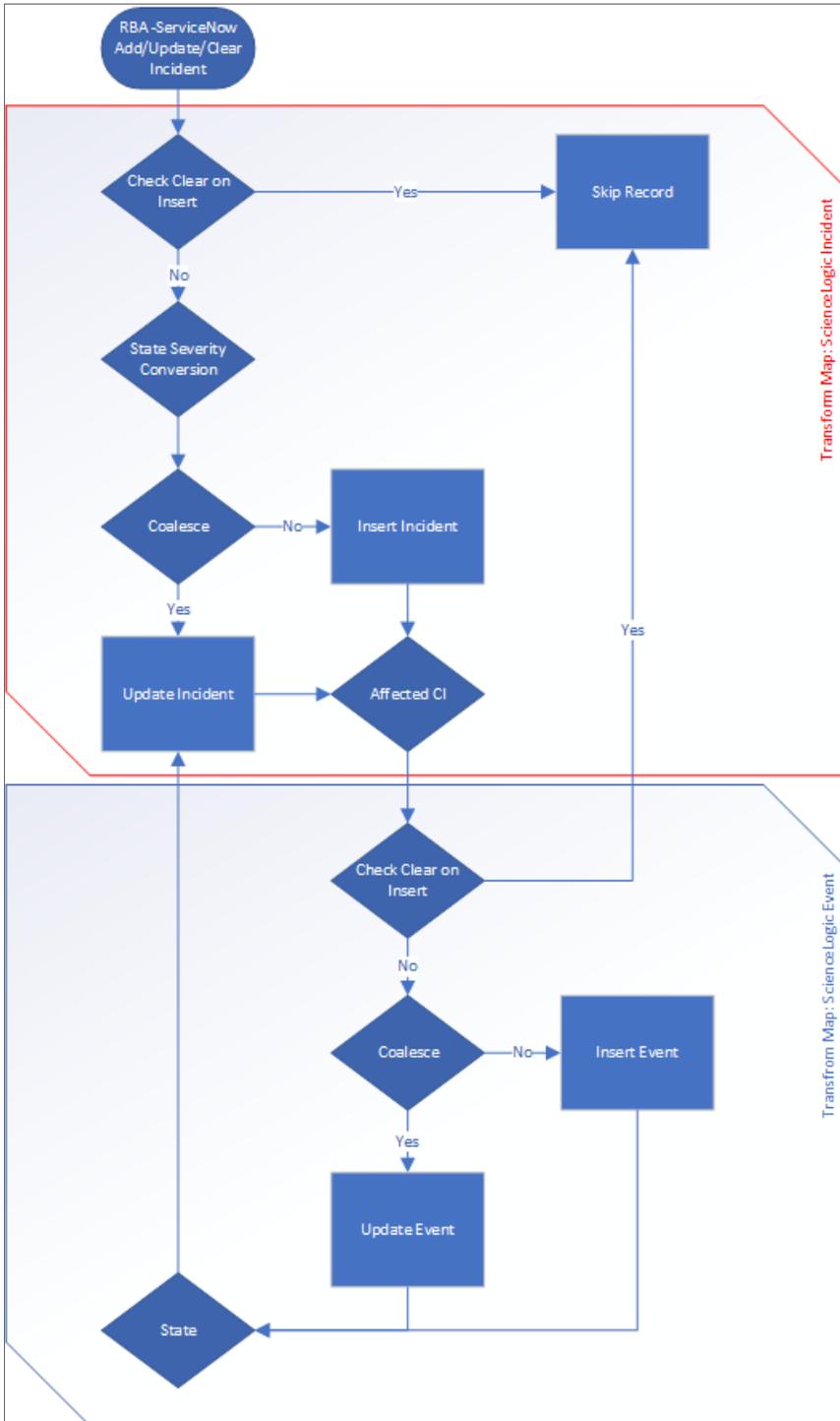
1. Through a Run Book Automation, SL1 identifies an event that should be synced to ServiceNow.
2. A Run Book Action executes a POST action to the Integration Service API to let the Integration Service know that an integration should be run to sync the event.
 - If the Run Book Action is successful and the POST responds with a 200, then the event data is stored in the Integration Service queue for syncing.
 - If the POST does not respond with a 200, then the Run Book Action inserts the missed event into a table in the SL1 database so that it can be retrieved later.
3. In parallel, a scheduled Integration Service event continuously checks the SL1 database for any missed events. If any missed events are found, they will be pulled from the database and inserted into the Integration Service queue.

NOTE: The Integration Service queue is persistently saved to disk, so if the service stops, any events that existed in the queue before to the failure will still exist in the queue after the service is running again.

4. Missed events are not removed from the SL1 database until after they are inserted into the queue.

Overview of the Run Book Automation for Incident Sync

You can configure a Run Book Automation to ensure that whenever SL1 detects a new, acknowledged, or cleared event, a corresponding incident is created or updated in ServiceNow:



The "ServiceNow: Add/Update/Clear Incident" Run Book Action policy is responsible for sending the SL1 payload to the Integration Service. The Integration Service then sends that payload to ServiceNow and creates, updates, acknowledges, or clears an incident, as needed.

SL1 features three Run Book Automation policies that facilitate this process:

- ServiceNow: Add/Update Incident
- ServiceNow: Event Acknowledged
- ServiceNow: Event Cleared

NOTE: A fourth Run Book Automation policy, "ServiceNow: Click to Create Incident" lets you manually create an incident in ServiceNow by clicking the life-preserver icon (🛟) in SL1. For more information, see [Configuring the "ServiceNow: Click to Create Incident" Automation Policy](#).

NOTE: The "Sync Incident State from ServiceNow to SL1 Event" integration application does not have an associated Run Book Action that triggers Incident Sync. You must schedule this integration application to run every minute, or to a time suitable for your requirements. You can use a cron job to trigger this schedule, or you can use the Integration Service user interface to schedule the application. For more information about scheduling, see .

Each Run Book Automation policy calls a single action in SL1. Ensure that the integration application in the Integration Service points to the relevant SL1 system and ServiceNow instance. The action then calls an integration application on the Integration Service that determines the workflow to execute.

Events in SL1 frequently occur and resolve due to fluctuations in the network and other changing conditions. However, the Run Book Automation policies above use a de-duplication algorithm to ensure that only a single open ServiceNow incident exists per device. As a result, if a device already has an existing ServiceNow incident, the following updates are made to the ServiceNow incident record:

- The "Work Notes" activity log in the incident record is updated with information about the secondary event(s).
- If a secondary event is of a higher severity than the event that originally created the ServiceNow incident, then the **Impact**, **Urgency**, and **Priority** fields are updated automatically in the ServiceNow incident record. If the secondary event is of a lesser severity, those fields are not updated.
- If an event is cleared in SL1 and then later reoccurs before the incident has been "Closed" in ServiceNow, then the subsequent events appear in the original ServiceNow incident record for that device. If an incident record has been "Closed," then ServiceNow will create a new incident record when a cleared event reoccurs in SL1.
- By default, if an event is acknowledged in SL1, the ServiceNow incident record will be updated with the work notes and the acknowledging user. Clearing an SL1 event will move the ServiceNow incident record state to "Resolved". If all SL1 events associated with a ServiceNow incident record are clear, the ServiceNow incident record will, by default, move to a "Resolved" state.

NOTE: You can edit the Run Book Action Snippet code to adjust the behavior for changing states when an SL1 event is acknowledged or cleared.

Configuring Incident Sync

This workflow covers how to set up the Run Book Automation to run Incident Sync:

1. [Create a configuration object in the Integration Service user interface](#)
2. [Align the new configuration file with the following integration applications:](#)
 - o "Create or Update ServiceNow Incident from SL1 Event"
 - o "Sync Incident State from ServiceNow to SL1 Event"
 - o "Update ServiceNow Incident when SL1 Event is Acknowledged"
 - o "Update ServiceNow Incident when SL1 Event is Cleared"
3. [Create a ServiceNow credential in SL1](#)
4. [Enable the following Run Book Automation Policies:](#)
 - o "ServiceNow: Add/Update Incident"
 - o "ServiceNow: Event Acknowledged"
 - o "ServiceNow: Event Cleared"
3. [Enable and customize the "ServiceNow: Add/Update/Clear Incident" Run Book Action Policy](#)
4. Optionally, [send custom data to ServiceNow using the Passthrough Option](#)
5. Optionally, [enable and configure the "ServiceNow: Click to Create Incident" Policy](#)

Creating and Aligning a Configuration Object

A **configuration object** supplies the login credentials to execute the steps for various integration applications in the Integration Service. You can use more than one configuration object as needed, and the **Configurations** page of the Integration Service user interfaces lists all available configuration objects for that system.

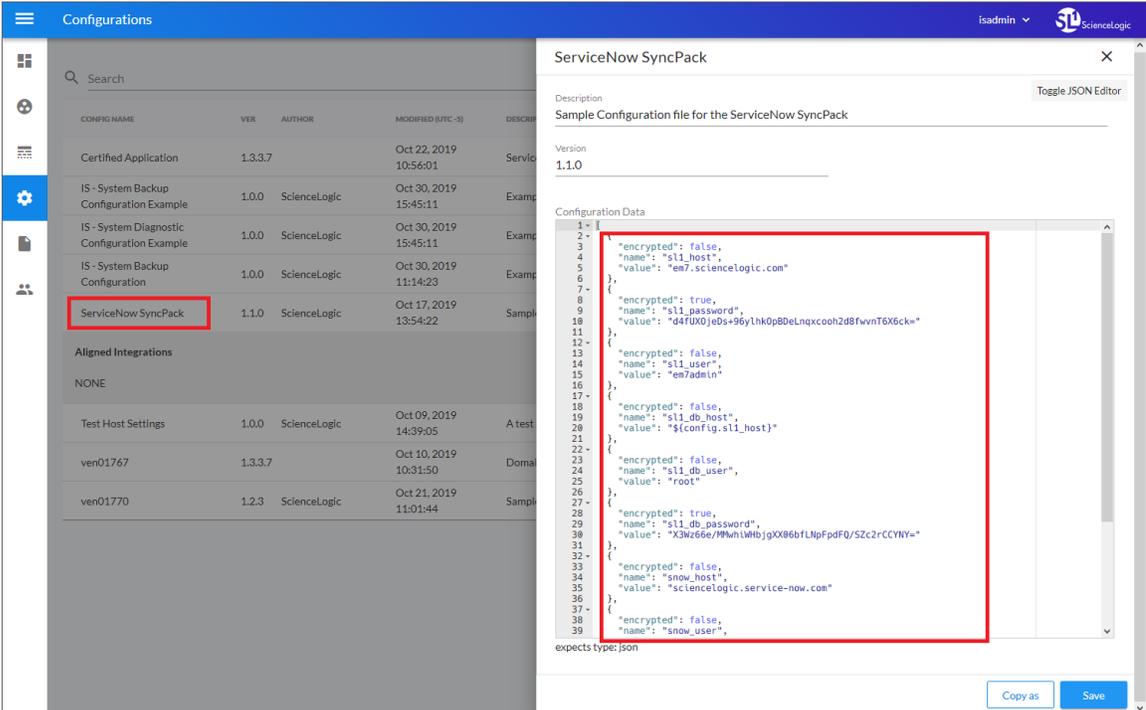
To use Incident Sync, you will need to create a new configuration object or file in the Integration Service user interface and align that configuration object to the relevant integration applications that are triggered by the Run Book Actions in SL1.

Creating a Configuration Object

For this SyncPack, you can make a copy of the "ServiceNow SyncPack" configuration object, which is the sample configuration file that was installed with the *ServiceNow Base SyncPack*.

To create a new configuration object based on the "ServiceNow SyncPack" configuration object:

1. In the Integration Service user interface, go to the **Configurations** page (⚙️).
2. Click the **[Actions]** (⋮) for the "ServiceNow SyncPack" configuration object and select *Edit*. The **Configuration** pane appears:



The screenshot shows the ServiceNow Configurations page with a table of configuration objects. The 'ServiceNow SyncPack' object is highlighted with a red box. The configuration details pane is open, showing the JSON configuration data for the 'Sample Configuration file for the ServiceNow SyncPack'.

CONFIG NAME	VER	AUTHOR	MODIFIED (UTC-9)	DESCRIPTION
Certified Application	1.3.3.7		Oct 22, 2019 10:56:01	Service
IS - System Backup Configuration Example	1.0.0	ScienceLogic	Oct 30, 2019 15:45:11	Exampl
IS - System Diagnostic Configuration Example	1.0.0	ScienceLogic	Oct 30, 2019 15:45:11	Exampl
IS - System Backup Configuration	1.0.0	ScienceLogic	Oct 30, 2019 11:14:23	Exampl
ServiceNow SyncPack	1.1.0	ScienceLogic	Oct 17, 2019 13:54:22	Sampl

Configuration Data

```
1- {
2-   "encrypted": false,
3-   "name": "sli_host",
4-   "value": "em7.sciencelogic.com"
5- },
6- {
7-   "encrypted": true,
8-   "name": "sli_password",
9-   "value": "d4fUX0jE8s+96y1hkOpB0eLmqxcooh2d8fwnt6X6ck="
10- },
11- {
12-   "encrypted": false,
13-   "name": "sli_user",
14-   "value": "ca7admin"
15- },
16- {
17-   "encrypted": false,
18-   "name": "sli_db_host",
19-   "value": "${config.sli_host}"
20- },
21- {
22-   "encrypted": false,
23-   "name": "sli_db_user",
24-   "value": "root"
25- },
26- {
27-   "encrypted": true,
28-   "name": "sli_db_password",
29-   "value": "X3wz6de/W9wLwWbjgX086bFLNpFpFQ/SZc2rCCWY="
30- },
31- {
32-   "encrypted": false,
33-   "name": "snow_host",
34-   "value": "sciencelogic.service-now.com"
35- },
36- {
37-   "encrypted": false,
38-   "name": "snow_user",
39- }
```

TIP: If needed, click **[Toggle JSON Editor]** to show the JSON code in the **Configuration** pane.

- Copy all of the sample configuration data from the **Configuration Data** field, and close the **Configuration** pane. Click [**Copy as**]. The **Create Configuration** pane appears:

The screenshot shows a 'Create Configuration' dialog box with a close button (X) in the top right corner. A 'Toggle JSON Editor' button is located in the top right area. The form contains the following fields:

- Friendly Name**: A text input field.
- Description**: A text input field.
- Author** and **Version**: Two text input fields.
- Configuration Data Values**: A table with 8 rows, each containing a Name, Value, an Encrypted checkbox, and a close button (X).

Name	Value	Encrypted	
sl1_host	em7.sciencelogic.com	<input type="checkbox"/>	X
sl1_password	SVh9nTCUEffef1edepEljW8NrgF	<input checked="" type="checkbox"/>	X
sl1_user	em7admin	<input type="checkbox"/>	X
sl1_db_host	\${config.sl1_host}	<input type="checkbox"/>	X
sl1_db_user	root	<input type="checkbox"/>	X
sl1_db_password	IDPx5lwLIX5HOQzsg6niL4FpSgc	<input checked="" type="checkbox"/>	X
snow_host	sciencelogic.service-now.com	<input type="checkbox"/>	X

A blue **Save** button is located at the bottom right of the dialog box.

- Complete the following fields:

- **Friendly Name**. Name of the configuration object that will display on the **Configurations** page.
- **Description**. A brief description of the configuration object.
- **Author**. User or organization that created the configuration object.
- **Version**. Version of the configuration object.

5. Paste the code you copied from the **ServiceNow SyncPack** configuration object into the **Configuration Data** field. If needed, click **[Toggle JSON Editor]** to show the JSON code.
6. In the **Configuration Data** field, be sure to include the required block of code to ensure that the integration applications aligned to this configuration object do not fail:

```
{
  "encrypted": false,
  "name": "s11_db_host",
  "value": "${config.s11_host}"
}
```

For example:

```
{
  "encrypted": false,
  "name": "s11_db_host",
  "value": "10.2.11.42"
}
```

NOTE: If you are using SL1 with an External Database (SL1 Extended architecture or a cloud-based architecture), update the "value" of that block of code to be the host of your database. This field accepts IP addresses. For example: "value": "db.sciencelogic.com". If you are *not* using the SL1 Extended architecture or a cloud-based architecture, you do not need to make any changes to the block of code other than pasting the code into the configuration object.

7. When editing or creating a configuration variable in the JSON editor, note the syntax:
 - The configuration file is surrounded by square brackets.
 - Each variable definition is surrounded by curly braces.
 - Each key name is surrounded by double-quotes and followed by a colon, while each value is surrounded by double-quotes and followed by a comma.
 - Each key:value pair in the definition is separated with a comma after the closing curly brace. The last key:value pair should not include a comma.
8. To edit or create a configuration variable, define the following keys:
 - **encrypted**. Specifies whether the value will appear in plain text or encrypted in this JSON file. If you set this to "true", when the value is uploaded, the Integration Service encrypts the value of the variable. The plain text value cannot be retrieved again by an end user. The encryption key is unique to each Integration Service system. The value is followed by a comma.
 - **name**. Specifies the name of the configuration file, without the JSON suffix. This value appears in the user interface. The value is surrounded by double-quotes and followed by a comma.
 - **value**. Specifies the value to assign to the variable. The value is surrounded by double-quotes and followed by a comma.

9. In the **Configuration Data** field, update the default variable definitions to match your Integration Service configuration. For this SyncPack, you must define the following variables in your new configuration object:

- *sl1_host*
- *sl1_password*
- *sl1_user*
- *sl1_db_user*
- *sl1_db_password*
- *snow_host*
- *snow_user*
- *snow_password*
- *region* (this is a user-defined variable that will identify your SL1 instance within ServiceNow)

TIP: The **ServiceNow SyncPack** configuration object contains all of the required variables. Simply update the variables to match your SL1 and ServiceNow settings.

10. Click the **[Save]** button.

NOTE: In a step, you can include the **config.** prefix with a variable to tell the Integration Service system to look in a configuration object to resolve the variable.

Aligning a Configuration Object

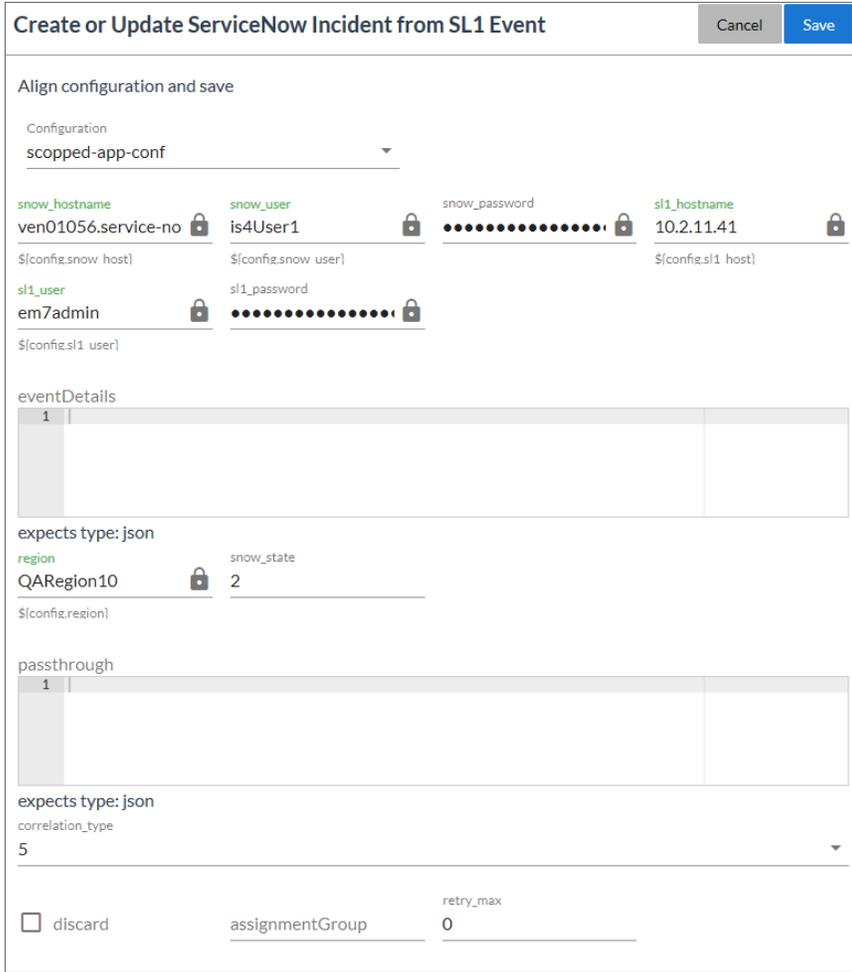
To run Incident Sync, you must "align" the configuration object to run with the following integration applications:

- "Create or Update ServiceNow Incident from SL1 Event"
- "Update ServiceNow Incident when SL1 Event is Acknowledged"
- "Update ServiceNow Incident when SL1 Event is Cleared"
- "Sync Incident State from ServiceNow to SL1 Event"

NOTE: The "Sync Incident State from ServiceNow to SL1 Event" integration application is the *only* application for Incident Sync that can be run manually or scheduled. The other three applications should only be triggered by Run Book Automations.

To align the configuration object with the relevant integration applications:

1. From the **Integrations** page of the user interface, open an "incident" integration application and click **[Configure]** (). The **Configurations** pane for that application appears:



Create or Update ServiceNow Incident from SL1 Event Cancel Save

Align configuration and save

Configuration
scoped-app-conf

snow_hostname: ven01056.service-no  snow_user: is4User1  snow_password:   sl1_hostname: 10.2.11.41 

\$(config.snow host) \$(config.snow user) \$(config.sl1 host)

sl1_user: em7admin  sl1_password:  

\$(config.sl1 user)

eventDetails

1

expects type: json

region: QARegion10  snow_state: 2

\$(config.region)

passthrough

1

expects type: json

correlation_type: 5

discard assignmentGroup retry_max: 0

2. From the **Configurations** drop-down, select the configuration object you created and click **[Save]** to align that configuration with the integration application.

NOTE: The values for **eventDetails** and the other parameters that appear in the **Configuration** pane with a padlock icon () are populated either by the configuration object you aligned with the integration application or by the Run Book Action. Do not modify these values. If you encounter an error, make sure your Run Book Action is configured properly.

Creating a ServiceNow Credential in SL1

To configure SL1 to communicate with ServiceNow, you must first create a SOAP/XML credential. This credential allows the Run Book Automation scripts and the Dynamic Applications in the *ServiceNow Base Pack PowerPack* to connect with your ServiceNow instance. These Run Book Automations are responsible for sending the SL1 event data to the Integration Service, which ultimately sends the data to a ServiceNow Incident.

The **ServiceNow RBA - Example** credential from the *ServiceNow Base Pack PowerPack* is an example SOAP/XML credential that you can configure for your own use.

To configure the **ServiceNow RBA - Example** credential:

1. In SL1, go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **ServiceNow RBA - Example** credential, then click its wrench icon (🔧). The **Edit SOAP/XML Credential** page appears:

The screenshot shows the 'Edit SOAP/XML Credential #85' window. It contains the following fields and sections:

- Basic Settings:** Profile Name (ServiceNow RBA - Example), Content Encoding ([text/xml]), Method ([POST]), HTTP Version ([HTTP/1.1]), URL [http(s)://Host:Port/Path | %D = Aligned Device Address | %N = Aligned Device Host Name] (https://INTEGRATIONSERVICEHOSTNAME), HTTP Auth User (USERNAME), HTTP Auth Password (masked), Timeout (seconds) (5).
- Soap Options:** Embedded Password [%P], Embed Value [%1], Embed Value [%2], Embed Value [%3], Embed Value [%4].
- Proxy Settings:** Hostname/IP, Port (0), User.
- CURL Options:** A list of options (CAINFO, CAPATH, CLOSEPOLICY, CONNECTTIMEOUT, COOKIE, COOKIEFILE, COOKIEJAR, COOKIELIST, CRLF, CUSTOMREQUEST, DNSCACHETIMEOUT) and a large empty text area.

Buttons: 'New', 'Reset' (top right); 'Save', 'Save As' (bottom).

3. Complete the following fields:
 - **Profile Name.** Type a new name for the ServiceNow credential.
 - **Content Encoding.** Make sure *text/xml* is selected.
 - **Method.** Make sure *POST* is selected.
 - **HTTP Version.** Select *HTTP/1.1*.
 - **URL.** Type the URL for your Integration Service instance.
 - **HTTP Auth User.** Type the username of your Integration Service instance.

- **HTTP Auth Password**. Type the password of your Integration Service instance.
- **Timeout**. Type "5".

4. Click **[Save As]**.
5. When the confirmation message appears, click **[OK]**.
6. On the **Credential Management** page (System > Manage > Credentials), make a note of the value in the **ID** column for the credential you just created:

Profile Name	Organization	RQ Use	RW Use	DA Use	Type	Credential User	Host	Port	Timeout (ms)	ID	Last Edited	Edited By
ServiceNow RBA - Example	[all orgs]	--	--	--	SOAP/XML Host USERNAME	INTEGRATIONSERVICEHOSTNAME		443	5000	107	2019-07-15 11:10:54	em7admin

You will use this value with the "sl1_credential_id" parameter when you *enable and customize the snippet code of the "ServiceNow: Add/Update/Clear Incident" Run Book Action*:

Policy Editor | Editing Action [50]

Action Name: ServiceNow: Add/Update/Clear Incident

Action State: [Enabled]

Description: Adds and Updates Incidents in ServiceNow.

Organization: [System]

Action Type: ServiceNow: Create, Update, Clear Incident (1.0)

Execution Environment: [-- Default: ServiceNow Base Pack]

Action Run Context: [Database]

Input Parameters:

```
{
  "sl1_credential_id": "",
  "debug": false,
  "configuration": "",
  "queue": "",
  "discard_if_no_ci": false,
  "is_app_name_new": "incident_sync_update_create",
  "is_app_name_ack": "event_acknowledged",
  "is_app_name_clear": "event_cleared",
  "servicenow_state_new": "1",
  "servicenow_state_ack": "",
  "servicenow_state_clear": "6",
  "correlation_type": "5",
  "assignment_group_new": "",
  "assignment_group_ack": ""
}
```

Buttons: Save, Save As

Enabling the Run Book Automation Policies

Before you can run the "ServiceNow: Add/Update/Clear Incident" Run Book Action, you must enable the three related Run Book Automation policies in SL1 :

- ServiceNow: Add/Update Incident
- ServiceNow: Event Acknowledged
- ServiceNow: Event Cleared

To enable the three ServiceNow Run Book Automation policies:

1. In SL1 , go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Locate the "ServiceNow: Add/Update Incident" automation policy and click its wrench icon (🔧). The **Automation Policy Editor** page appears:

The screenshot shows the 'Automation Policy Editor' interface for editing the 'ServiceNow: Add/Update Incident' policy. The interface is divided into several sections:

- Policy Configuration:** Includes fields for Policy Name, Policy Type (Active Events), Policy State (Enabled), Policy Priority (High), and Organization (System). The Policy State and Policy Priority fields are highlighted with red boxes.
- Criteria Logic:** A series of dropdown menus for defining event criteria, such as severity and time conditions.
- Match Logic:** A dropdown menu for defining the match logic (e.g., Text search).
- Repeat Time:** A dropdown menu for defining the repeat time (e.g., Only once).
- Align With:** A dropdown menu for defining the alignment (e.g., Devices).
- Trigger on Child Rollup:** A checkbox that is currently checked.
- Available Devices:** A list of devices, including 'System' and several ScienceLogic EM7 All-In-One devices.
- Aligned Devices:** A list containing '(All devices)'. Navigation arrows are present between the available and aligned lists.
- Available Events:** A list of events, including various critical alerts from AKCP sensors.
- Aligned Events:** A list containing '(All events)'. Navigation arrows are present between the available and aligned lists.
- Available Actions:** A list of actions, including 'SNMP Trap [1]: EM7 Event Trap' and several 'Snippet [5]: AWS: ...' actions.
- Aligned Actions:** A list containing '1. ServiceNow: Create, Update, Clear Incident [100]: Sr'. Navigation arrows are present between the available and aligned lists.

At the bottom of the editor, there are 'Save' and 'Save As' buttons.

3. Update the following fields:
 - **Policy State.** Select *Enabled*.
 - **Policy Priority.** Select *High* to ensure that this Integration Service automation policy is added to the top of the queue.
 - **Available Actions.** If it is not already selected, select the corresponding ServiceNow Run Book Action policy.

TIP: By default, the "ServiceNow: Add/Update Incident" automation policy will create ServiceNow incidents for **all** devices. You can limit the devices affected by making changes to the **Organization, Severity, Match Logic, Aligned Devices,** and/or **Aligned Events** fields.

WARNING: ScienceLogic highly recommends that you do not make changes to the **Policy Type, Repeat Time,** or **Align With** fields or the *And event is NOT acknowledged* setting.

4. Click **[Save]**.
5. Repeat steps 2-4 for the "ServiceNow: Event Acknowledged" and "ServiceNow: Event Cleared" Run Book Automation policies.

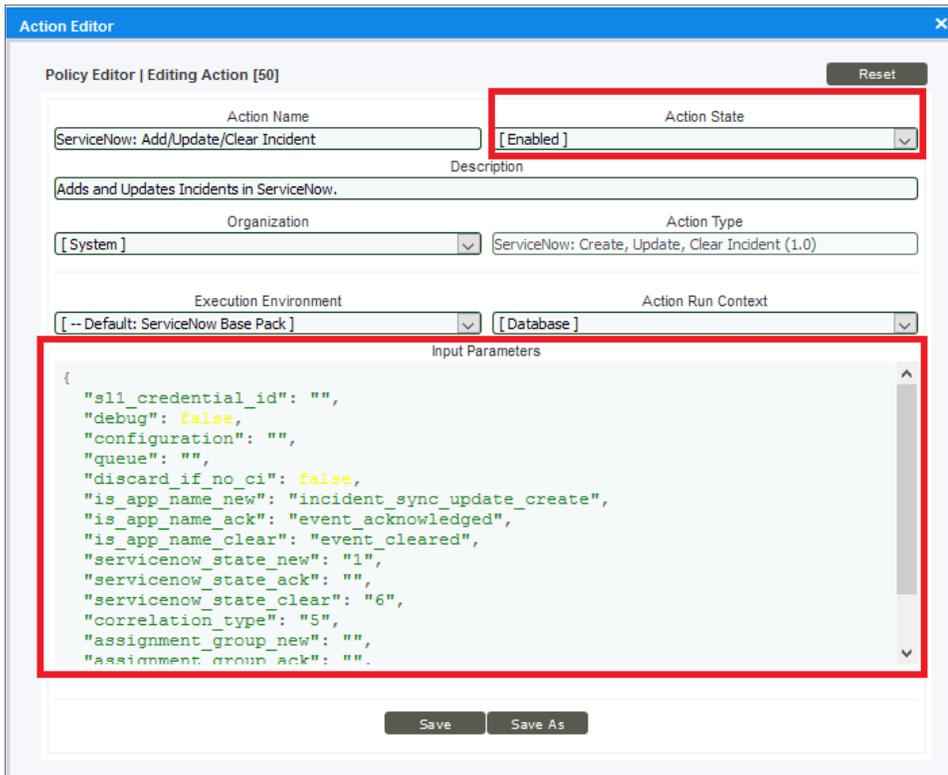
Enabling and Customizing the Run Book Action Policy

The "ServiceNow: Add/Update/Clear Incident" Run Book Action policy contains several default values in the snippet code for the policy that you can customize for Incident Sync. You can edit these values in the **Input Parameters** pane of the **Action Policy Editor** page for this policy.

For example, the **Correlation Type** is a value that you can use to correlate an SL1 event with a ServiceNow incident. For the Correlation Type, which is also called the "Correlation ID", you can choose a number of different ways in which an incident can be created. The default setting is "correlation_type": "5", which means that this Run Book Action correlates all events by device ID and event policy ID, and if the event matches and the state is active, the Action updates the existing incident. The Action creates a new incident if the event does *not* match by device ID and event policy ID. As a result, the Correlation Type helps determine which events get rolled up under an incident. All Run Book Action scripts should use the same Correlation Type value, otherwise correlation will fail.

To enable and customize the Run Book Automation policies:

1. In SL1, go to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. Locate the **ServiceNow: Add/Update/Clear Incident** policy and click its wrench icon (🔧). The **Action Policy Editor** page appears:



The screenshot shows the 'Action Editor' window with the following fields and values:

- Action Name:** ServiceNow: Add/Update/Clear Incident
- Action State:** [Enabled]
- Description:** Adds and Updates Incidents in ServiceNow.
- Organization:** [System]
- Action Type:** ServiceNow: Create, Update, Clear Incident (1.0)
- Execution Environment:** [-- Default: ServiceNow Base Pack]
- Action Run Context:** [Database]
- Input Parameters:**

```
{
  "sli_credential_id": "",
  "debug": false,
  "configuration": "",
  "queue": "",
  "discard_if_no_ci": false,
  "is_app_name_new": "incident_sync_update_create",
  "is_app_name_ack": "event_acknowledged",
  "is_app_name_clear": "event_cleared",
  "servicenow_state_new": "1",
  "servicenow_state_ack": "",
  "servicenow_state_clear": "6",
  "correlation_type": "5",
  "assignment_group_new": "",
  "assignment_group_ack": ""
}
```

Buttons at the bottom: Save, Save As, and a Reset button in the top right corner.

3. For the **Action State** field select *Enabled*.
4. In the **Input Parameters** pane, edit the snippet code as necessary, using the information in the **Customizing the Snippet Code in the Input Parameters Pane** section, below. When you are finished, click **Save**.

Customizing the Snippet Code in the Input Parameters Pane

SL1 Run Book Action snippets are written in Python. In the event of a syntax error, the policies will no longer run. As a result, you must ensure that all edits adhere to Python standards. True and False options are case-sensitive and must not contain quotes.

NOTE: Previous SyncServer users had three separate Run Book Action scripts for add/update, acknowledge, and clear. These have been rolled into a single Run Book Action in the Integration Service, but there are still three Automation policies.

You can customize the following values in the "ServiceNow: Add/Update/Clear Incident" Run Book Action snippet code:

- **sl1_credential_id**. Specifies the ID of the credential object. You can find this value in the **ID** column of the **Credentials** page (System > Manage > Credentials of SL1).
- **debug**. A true/false value that determines if the action is logged in SL1 and if the application is run in Debug Mode on the Integration Service. Troubleshooting logs are written to **/data/tmp/servicenow_rba.log**.
- **configuration**. Specifies the ID of the configuration object used on the Integration Service. The configuration ID is all lower-case, with spaces in the configuration object "friendly" name replaced by underscores. For example: "configuration": "servicenow_syncpack_configs"

NOTE: To find the configuration ID with the API, make a GET request on this endpoint:
https://<Integration_Service_hostname>/api/v1/configurations.

- **queue**. Specifies the worker queue on which the application runs. Leave this as default.
- **discard_if_no_ci**. Specifies whether the Integration Service should create incidents in ServiceNow for devices that do not have a matching CI record. The default is *false*. Your options include:
 - *true*. If a device is not mapped to a CI, the Integration Service will *not* create an incident in ServiceNow, and SL1 is not updated. The following log message appears: "No CI found".
 - *false*. If a device is not mapped to a CI, the Integration Service *will* create an incident in ServiceNow and update SL1.
- **servicenow_state_new**:
 - 1. Incident state is "New". This is the default value.
 - 2. Incident state is "In Progress".
 - 3. Incident state is "On Hold".
 - 6. Incident state is "Resolved".
 - 7. Incident state is "Closed".
 - 8. Incident state is "Canceled".
- **servicenow_state_ack**:
 - 1. Incident state is "New". There is no default value.
 - 2. Incident state is "In Progress".
 - 3. Incident state is "On Hold".
 - 6. Incident state is "Resolved".
 - 7. Incident state is "Closed".
 - 8. Incident state is "Canceled".

- **servicenow_state_clear:**
 - 1. Incident state is "New".
 - 2. Incident state is "In Progress".
 - 3. Incident state is "On Hold".
 - 6. Incident state is "Resolved". This is the default value.
 - 7. Incident state is "Closed".
 - 8. Incident state is "Canceled".
- **correlation_type.** Correlates an SL1 event with a ServiceNow incident. For this value, which is also called the "Correlation ID", you can choose a number of different ways in which an incident can be created. The default setting is "correlation_type": "5", which means that this Run Book Action correlates all events by device ID and event policy ID, and if the event matches and the state is active, the Action updates the existing incident. The Action creates a *new* incident if the event does *not* match by device ID and event policy ID. As a result, the Correlation Type helps determine which events get rolled up under an incident. All Run Book Action scripts should use the same Correlation Type value, otherwise correlation will fail.

The possible Correlation Type values include the following:

- 1. Correlate all duplicate incidents by SL1 ID only.
 - 2. Correlate all duplicate incidents by event policy ID only.
 - 3. Correlate all duplicate incidents by device ID only.
 - 4. Correlate all duplicate incidents by Interface ID only. This correlation requires that the SL1 event has an interface aligned. If there is no interface aligned to the event, the returned Interface ID will be 0.
 - 5. Correlate all duplicate incidents by device ID and event policy ID. This is the default Correlation Type value.
 - 6. Correlate all duplicate incidents by device ID, event policy ID, and event sub entity ID.
 - 7. Correlate all duplicate incidents by device ID, event policy ID, region, and event ID. Please note that using this correlation type will result in all events triggering new incidents.
 - 8. Correlate all events to the root device ID of the component device into one incident. Please note that using this correlation type might lead to undesirable behavior, as not all events generated on a child device indicate a problem on the root device. As a result, ScienceLogic recommends that you only use this correlation type against specific event policies.
- You can assign the assignment group to one of the new, acknowledged, or cleared incidents that are mapped. To disable this feature, ensure that no values are set. After an incident is created, the assignment group value will not be changed by the Run Book Action. To assign an assignment group, set the variable value to the **sys_id** of the ServiceNow Assignment Group. In the following example, the assignment group is assigned to incidents that are *cleared*:

```
"assignment_group_new": "",
"assignment_group_ack": "",
"assignment_group_clear": "sys_id"
```

Customizing Logging in the Run Book Action

You can customize the following logging-related items in the "ServiceNow: Add/Update/Clear Incident" Run Book Action snippet code:

- `logfile = /data/tmp/ServiceNow_add_update_clear_incident.log`
 - Location for logging output.
 - Will be created if it does not exist.
 - Will be appended with each Run Book job.
 - Is case-sensitive.
- `do_debug_logging = True`
 - True is on, False is off.
 - Is case-sensitive.
 - For troubleshooting, these can be enabled or changed.
 - Writes logs to `/data/tmp/servicenow_rba.log`.

Sending Custom Data to ServiceNow Using the Passthrough Option

You can use the "ServiceNow: Add/Update Incident" Run Book Automation and the "ServiceNow: Add/Update/Clear Incident" Run Book Action to "pass through" custom data to ServiceNow. For example, you might want to use the passthrough functionality to overwrite the impact and urgency of a ServiceNow Incident, which is the only way to change the priority of the Incident.

To pass custom data to ServiceNow:

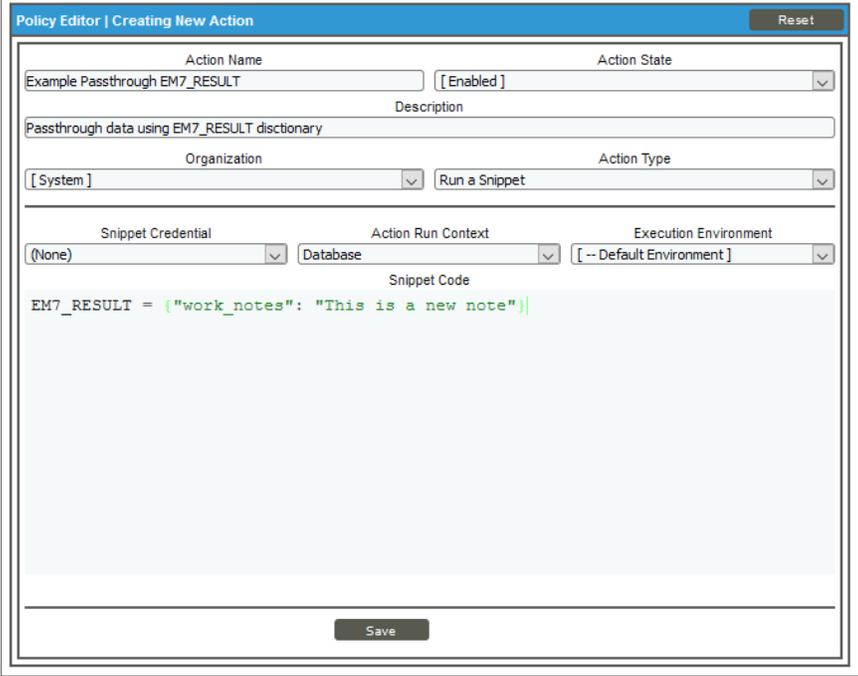
- Create a new Run Book Action that pulls the relevant data and adds it to a dictionary called `EM7_RESULT`.
- Add the new Run Book Action to the "ServiceNow: Add/Update Incident" Run Book Automation Policy, ahead of the "ServiceNow: Add/Update/Clear Incident" Run Book Action so that the new Action runs first, and then is consumed by the ServiceNow Action.

The following procedure describes how to configure the passthrough functionality.

Passing Custom Data to ServiceNow

To pass custom data to ServiceNow:

1. Go to the **Action Policy Manager** page (Registry > Run Book > Actions) and click **[Create]** to create a new Run Book Action policy:



The screenshot shows the 'Policy Editor | Creating New Action' interface. It features a 'Reset' button in the top right corner. The form includes several fields: 'Action Name' (text input with 'Example Passthrough EM7_RESULT'), 'Action State' (dropdown menu with '[Enabled]'), 'Description' (text input with 'Passthrough data using EM7_RESULT dictionary'), 'Organization' (dropdown menu with '[System]'), and 'Action Type' (dropdown menu with 'Run a Snippet'). Below these are 'Snippet Credential' (dropdown menu with '(None)'), 'Action Run Context' (dropdown menu with 'Database'), and 'Execution Environment' (dropdown menu with '[-- Default Environment]'). A 'Snippet Code' text area contains the code: `EM7_RESULT = {"work_notes": "This is a new note"}`. A 'Save' button is located at the bottom center of the form.

2. Complete the following fields:
 - **Action Name.** Type a unique name for the Action.
 - **Action State.** Select *Enabled*.
 - **Action Type.** Select *Run a Snippet*.
 - **Execution Environment.** Select *ServiceNow Base Pack*.
 - Complete the other fields as needed.

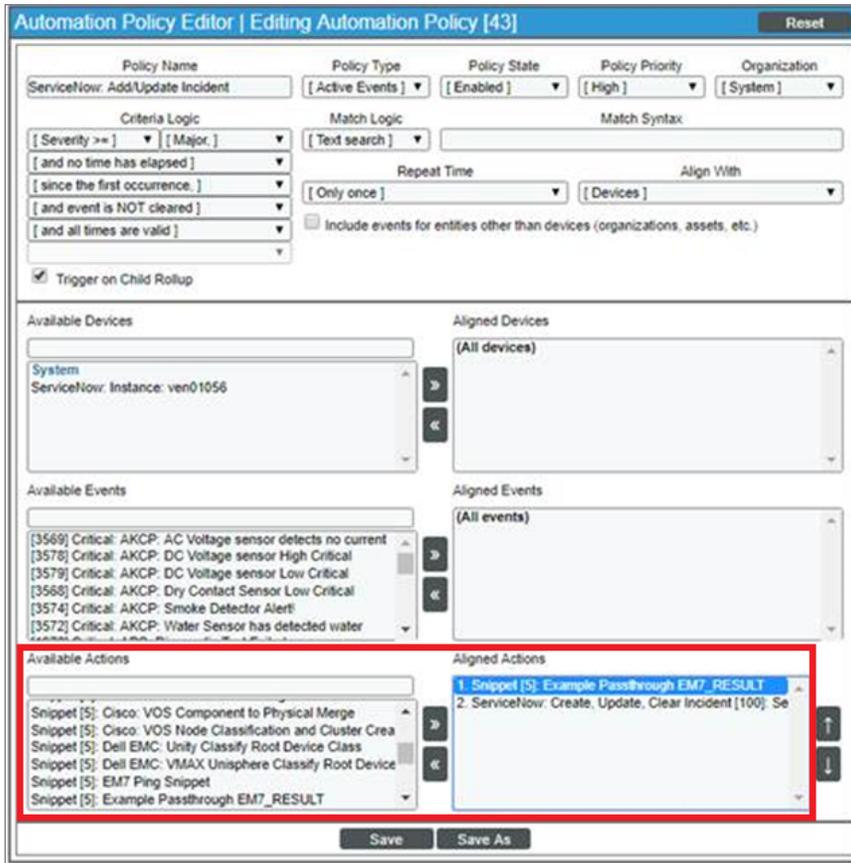
3. In the **Snippet Code** pane, add the code for the EM7_RESULT dictionary. For example, the following snippet code lets you override the ServiceNow Incident work notes with a hardcoded note:

```
EM7_RESULT = {"work_notes": "This is a new note"}
```

Additional notes about the structure of the EM7_RESULT dictionary:

- EM7_RESULT = is required for the dictionary, and the formatting of the keys should match the example above.
 - All keys defined in the EM7_RESULT dictionary need to map to field IDs on the **ScienceLogic Events** table in ServiceNow.
 - You can hard-code the values in the EM7_RESULT dictionary, or you can use variables and functions, like the [Snippet Code Example](#), below.
 - As a best practice, avoid sending null passthrough values to ServiceNow. If you must send 'null' or 'NULL' values to ServiceNow, pass through that value as an empty string, such as "location": "". Also, only pass through values that you need. For example, instead of sending {"location": "", "work_notes": "stuff"}, simply send {"work_notes": "stuff"}.
 - A long snippet might delay the ticket being created
4. Click **[Save]**.
 5. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation) and open the "ServiceNow: Add/Update Incident" Run Book Automation Policy.

- In the **Available Actions** section, add the new Run Book Action *before* the "ServiceNow: Create, Update, Clear Incident" Run Book Action:



NOTE: The output of this new Run Book Action will be consumed by the "ServiceNow: Create, Update, Clear Incident" Run Book Action, ensuring that the EM7_RESULT dictionary is passed through to ServiceNow. The "ServiceNow: Create, Update, Clear Incident" Run Book Action automatically populates the passthrough values with any values from EM7_LAST_RESULT. The passthrough overwrites any other previously defined fields, such as assignment group.

- You can add additional Run Book Actions to the Run Book Automation Policy for any additional workflows that you might want to run. The Automation Policy execute these Actions in a sequential, top-down order. However, the "ServiceNow: Create, Update, Clear Incident" Run Book Action only consumes the EM7_RESULT dictionary from the Run Book Action directly above it.

Snippet Code Example

The following snippet code example shows how to pull additional information and make it available for passthrough. All of the additional information that is going to be sent is contained in a dictionary variable called EM7_RESULT. You can pass through multiple items through in a single Run Book Action by adding additional keys to the EM7_RESULT dictionary.

This example lets you assign assignment groups to an Incident based on certain criteria, such as event policy IDs:

```
from future.utils import iteritems

def invert_mappings(mappings):
    """
    Invert received one-to-many mappings and converts it into a one-to-one
    mapping.

    Args:
        mappings (dict): Dictionary of mapped values

    Returns:
        dict: inverted dictionary.

    """
    inverted_mappings = dict()
    for key, values in iteritems(mappings):
        for sub_value in values:
            invert_mappings[sub_value] = key
    return inverted_mappings

# Example of assignment group to list of event policy ids mapping.
assignment_groups_to_event_policies = {
    "sys_id_1": [1, 2, 3, 4, 5],
    "sys_id_2": [6, 7, 8, 9, 10],
}
# which sys_id to use if the current event_policy_id isn't mapped
default_sys_id = "sys_id_3"

# invert the mappings
event_policy_to_assignment_group = invert_mappings(assignment_groups_to_event_
policies)

# Send assignment group sys_id to IS RBA
EM7_RESULT = {
    "assignment_group": event_policy_to_assignment_group.get(
        EM7_VALUES["%3"], default_sys_id
    )
}
```

Configuring the "ServiceNow: Click to Create Incident" Automation Policy

The "ServiceNow: Click to Create Incident" Run Book Automation policy lets you manually create an incident in ServiceNow by clicking the life-preserver icon (🛟) in SL1. This policy is available in the *ServiceNow Base Pack PowerPack*.

To configure the "ServiceNow: Click to Create Incident" policy:

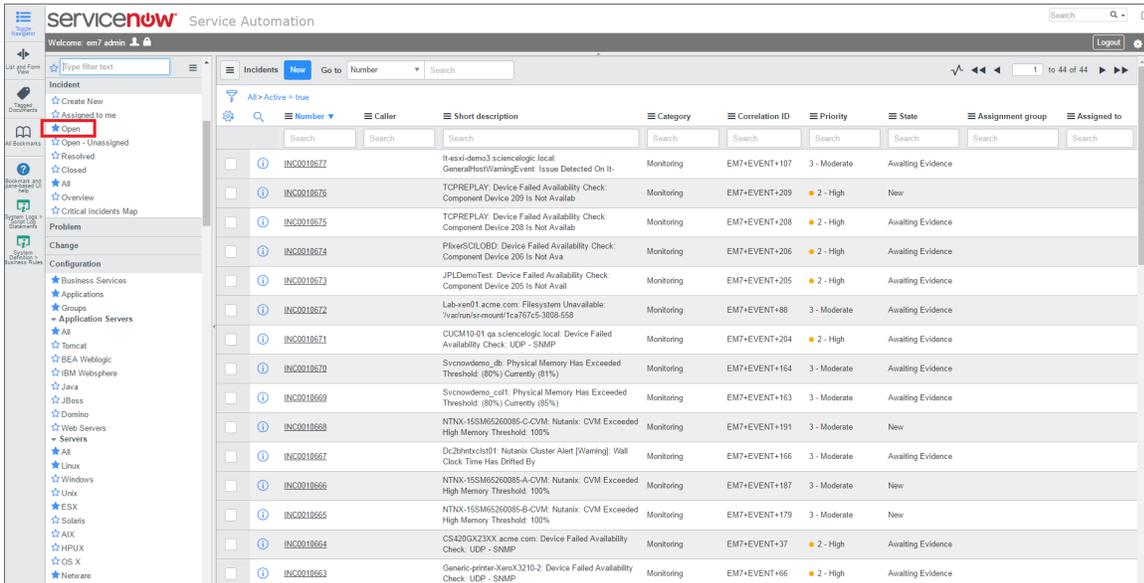
1. In SL1, go to the Behavior Settings page (System > Settings > Behavior) and set the **Event Console Ticket Life Ring Button Behavior** option to *Create/View External Ticket*.
2. Click **[Save]** to save your changes.
3. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
4. Locate the **ServiceNow: Click to Create Incident** policy and click its wrench icon (🔧). The **Automation Policy Editor** page appears:

The screenshot shows the 'Automation Policy Editor' interface for the policy 'ServiceNow: Click to Create Incident'. The 'Policy State' is set to 'Enabled'. The 'Repeat Time' is set to 'Every 1 minute until satisfied'. The 'Aligned Actions' list contains one action: '1. ServiceNow: Create, Update, Clear Incident [106]: Sr'. The interface also shows 'Available Devices', 'Available Events', and 'Available Actions' lists.

5. Update the following fields:
 - **Policy State**. Select *Enabled*.
 - **Repeat Time**. Specify the frequency at which SL1 should execute the automation policy while the conditions are still met. The choices range from "every 30 seconds until satisfied" to "every 2 hours until satisfied", or "only once". By default, the policy only runs once.
 - **Available Actions**. If it is not already selected, select the corresponding ServiceNow Run Book Action policy to add it to the **Aligned Actions** field.
6. Click **[Save]**.

Viewing Incidents and Events

To view SL1-created incidents in ServiceNow, go to the **Incidents** page (Incident > Open) in the ServiceNow application:



All SL1 incidents use the event message from the SL1 **Events** page as the incident description in ServiceNow. The SL1 **Events** also links to the Incident record if you click the life-preserver icon (🛟) under the **External Ticket** column. Note that you might need to enable the **External Ticket** column by modifying the Console preferences.

SL1 and ServiceNow use slightly different methods for designating the severity or priority of an event or incident. A custom data lookup table handles the conversion by translating the SL1 event severity into ServiceNow **Impact**, **Urgency**, and **Priority** fields. This severity data lookup table for automatically deploys with the "ScienceLogic SL1 : CMDB & Incident Automation Application", also called the Scoped or Certified Application:

Severity	Impact	Urgency
1	1	2
2	2	2
3	2	3
4	3	3

If a second event for the same incident occurs, that event will update the **Impact** and **Urgency** values, based on the severity lookup table.

When the SL1 Run Book Automation creates a ServiceNow incident, the action taken depends on the **Correlation Type** you specified in the "ServiceNow Add/Update/Clear Incident" Run Book Action. If one or more events are aligned to the same incident, those events will show up in the **[Events]** tabs at the bottom of the Incident record in ServiceNow. All event-specific information for an incident is listed on this page:

Task SLAs	Affected CIs	Impacted Services/CIs	Child Incidents	Events (13)					
Incident = INC0013377									
Action policy	Active	Automation policy	Correlation	Device	Event count	Event created	Event ID	Event policy	Hyperlink
ServiceNow: Add/Update/ Clear Incident	false	ServiceNow: Add/Update Incident	fsundemo89+DEV+2197+EVENT+1705	(empty)	1	2019-09-19 09:23:16	11765926	Poller: Network Latency Exceeded Threshold	http://em7.mydomain.com/em7/index.em?wsec=events&q_type=aid&q_arg=11765926&_sev=1&q_sort=0&q_oper=0
ServiceNow: Add/Update/ Clear Incident	false	ServiceNow: Add/Update Incident	fsundemo89+DEV+2197+EVENT+1705	(empty)	1	2019-09-19 08:28:17	11765576	Poller: Network Latency Exceeded Threshold	http://em7.mydomain.com/em7/index.em?wsec=events&q_type=aid&q_arg=11765576&_sev=1&q_sort=0&q_oper=0
ServiceNow: Add/Update/ Clear Incident	false	ServiceNow: Add/Update Incident	fsundemo89+DEV+2197+EVENT+1705	(empty)	1	2019-09-19 08:18:16	11765496	Poller: Network Latency Exceeded Threshold	http://em7.mydomain.com/em7/index.em?wsec=events&q_type=aid&q_arg=11765496&_sev=1&q_sort=0&q_oper=0
ServiceNow: Add/Update/ Clear Incident	false	ServiceNow: Add/Update Incident	fsundemo89+DEV+2197+EVENT+1705	(empty)	1	2019-09-19 07:33:17	11765206	Poller: Network Latency Exceeded Threshold	http://em7.mydomain.com/em7/index.em?wsec=events&q_type=aid&q_arg=11765206&_sev=1&q_sort=0&q_oper=0
ServiceNow: Add/Update/ Clear Incident	false	ServiceNow: Add/Update Incident	fsundemo89+DEV+2197+EVENT+1705	(empty)	1	2019-09-19 07:13:16	11765056	Poller: Network Latency Exceeded Threshold	http://em7.mydomain.com/em7/index.em?wsec=events&q_type=aid&q_arg=11765056&_sev=1&q_sort=0&q_oper=0
ServiceNow: Add/Update/ Clear Incident	false	ServiceNow: Add/Update Incident	fsundemo89+DEV+2197+EVENT+1705	(empty)	1	2019-09-19 07:03:18	11764986	Poller: Network Latency Exceeded Threshold	http://em7.mydomain.com/em7/index.em?wsec=events&q_type=aid&q_arg=11764986&_sev=1&q_sort=0&q_oper=0

TIP: If the [Events] tab does not display at the bottom of the Incident record, you can add it by opening the Incident record, clicking the Additional actions icon (☰), and selecting *Configure > Related Lists*. Add *Event > Incident* to the **Selected** column and click [Save].

For more information about ServiceNow incident management, see https://docs.servicenow.com/bundle/newyork-it-service-management/page/product/incident-management/concept/c_IncidentManagement.html.

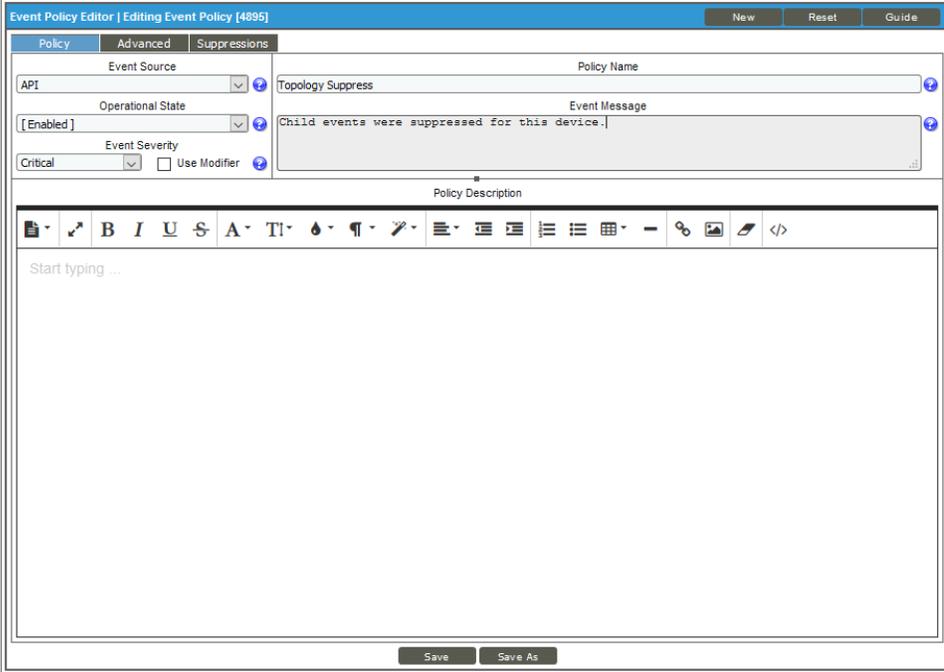
Incident Topology Suppression

Incident topology suppression is used when ServiceNow incidents that have been synced with SL1 devices occur on devices that have a parent/child relationship. If you choose to enable incident topology suppression in SL1, child events synced with ServiceNow incidents do not appear in the SL1 **Event Console** as separate events. Instead, the child events are nested under the parent event.

NOTE: The steps in this process use the Classic user interface for SL1.

To enable incident topology suppression:

1. In SL1, navigate to the **Event Policy Manager** page (Registry > Events > Event Manager) and click the **[Create]** button. The **Event Policy Editor** modal appears:



The screenshot shows the 'Event Policy Editor' modal window. The title bar reads 'Event Policy Editor | Editing Event Policy [4895]' and includes 'New', 'Reset', and 'Guide' buttons. The interface is divided into several sections:

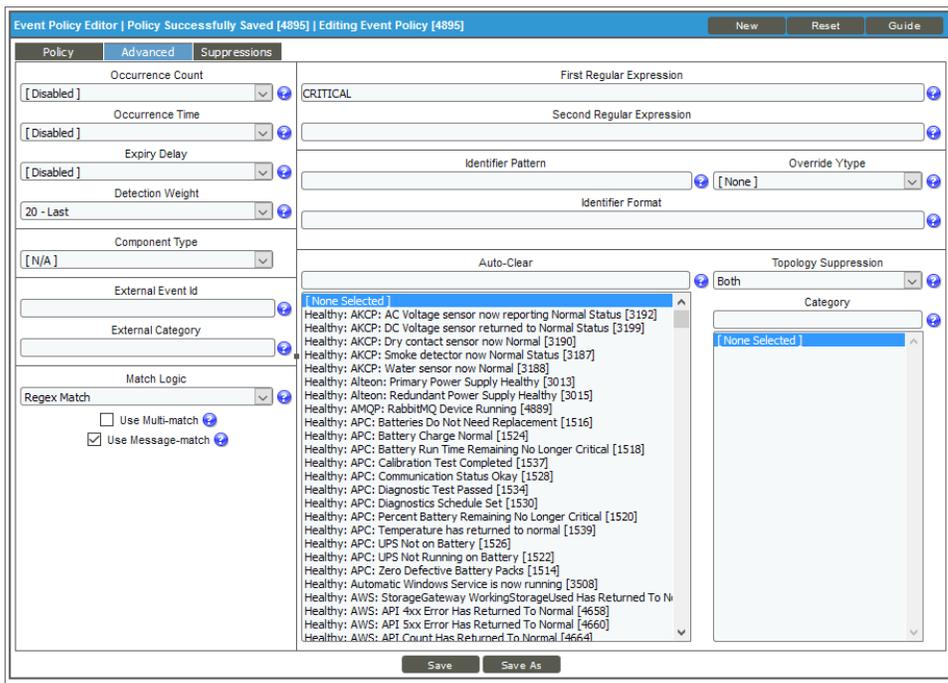
- Policy** tab (selected), with sub-tabs for 'Advanced' and 'Suppressions'.
- Event Source:** A dropdown menu set to 'API'.
- Operational State:** A dropdown menu set to '[Enabled]'.
- Event Severity:** A dropdown menu set to 'Critical' and an unchecked 'Use Modifier' checkbox.
- Policy Name:** A text input field containing 'Topology Suppress'.
- Event Message:** A text area containing 'Child events were suppressed for this device.'.
- Policy Description:** A large text area with a rich text editor toolbar (bold, italic, underline, strikethrough, text color, background color, bulleted list, numbered list, link, unlink, image, code) and the text 'Start typing ...'.

At the bottom of the modal are 'Save' and 'Save As' buttons.

2. On the **[Policy]** tab, update the following fields:

- **Event Source:** Select *API*.
- **Operational State:** Select *Enabled*.
- **Event Severity:** Select *Critical* as the severity of the event.
- **Policy Name.** Type the name of the event. Can be any combination of alphanumeric characters, up to 48 characters in length
- **Event Message.** Type the message that will appear when this event occurs.

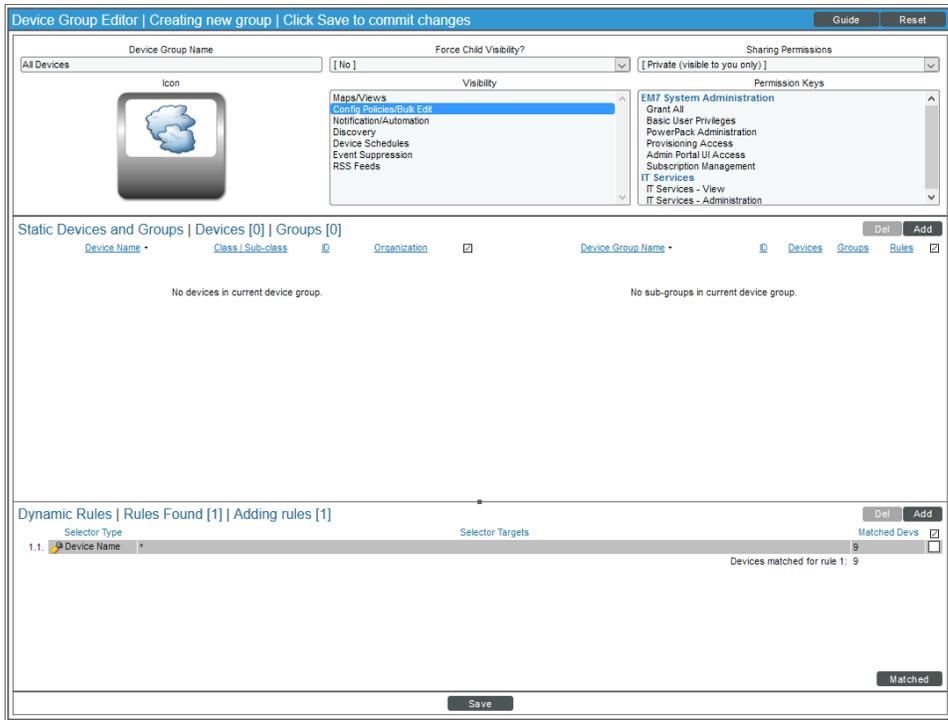
3. Click the **[Advanced]** tab.



4. On the **[Advanced]** tab, update the following fields:

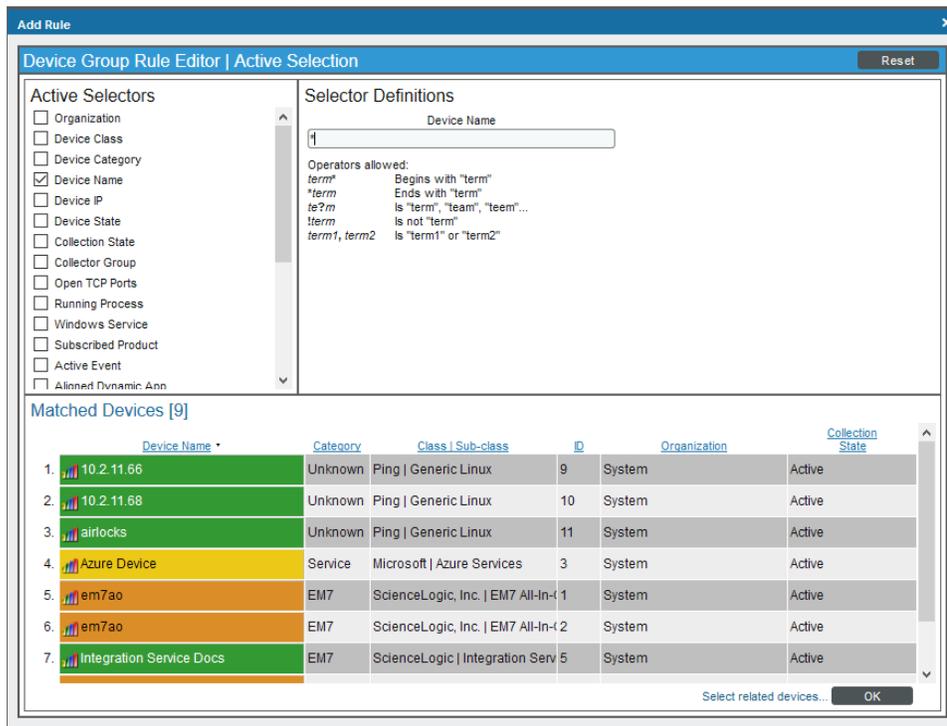
- **Detection Weight.** Select *20 - Last*. If two event definitions are very similar, the weight field specifies the order in which SL1 should match messages against the similar event definitions. The event definition with the lowest weight will be matched first. This field is most useful for events that use expression matching. Options range from 0 (first) - 20 (last).
- **Match Logic.** Select *Regex Match*. Specifies whether SL1 should process the First Match String field and Second Match String as regular expressions or as simple text matches. Because you selected *Regex Match*, you cannot define a "match all" expression by leaving the First Match String and Second Match String fields empty.
- **Use Message-match.** Select this option. If SL1 has generated an event and then a second log message or alert matches the same event policy for the same entity, SL1 will not generate a second event, but will increase the count value for the original event. This behavior will occur only if the log messages or alerts contain the same message.
- **First Regular Expression.** Type "CRITICAL" as the string used to correlate the event with a log message.
- **Topology Suppression.** Select *Both*. If this event occurs on a parent device, it behaves as a suppressing event. If this event occurs on a child device, it behaves as a suppressible event.

5. Click **[Save]** and close the **Event Policy Editor** modal.
6. Next, go to the **Device Groups** page (Registry > Device Groups) and click the **[Create]** button. A **Device Group Editor** page appears:



7. Complete the following fields, and leave the default settings for the remaining fields:
 - **Template Name.** Specify the name of the new device group.
 - **Force Child Visibility.** Select "No".
 - **Visibility.** Select *Config Policies/Bulk Edit* to let you configure all the devices in the new device group using a device template.

- Click the **[Save]** button and then click the **[Add]** button in the **Dynamic Rules** pane to add dynamic rules to the new device group. The Device Group Rule Editor modal page appears:

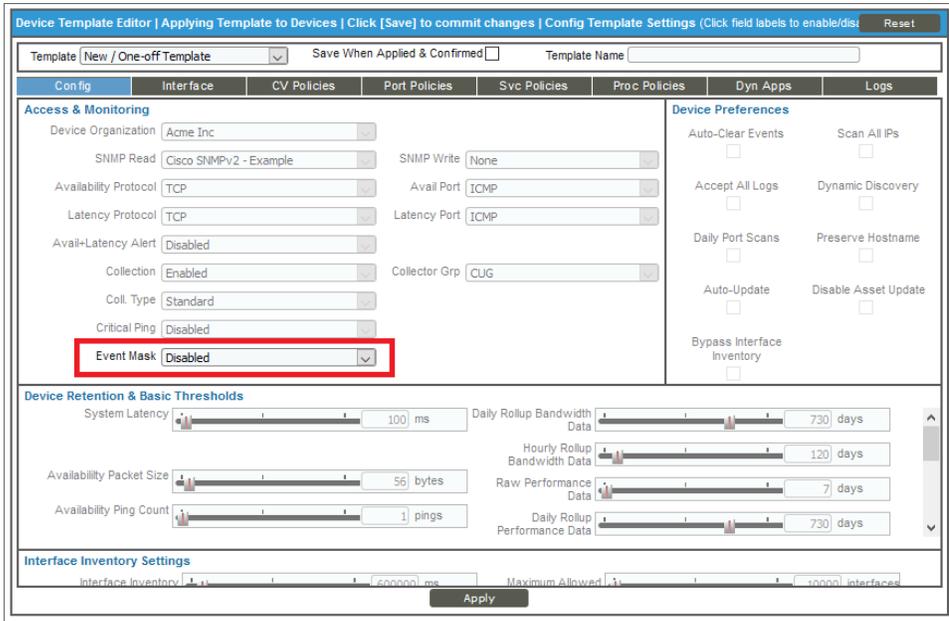


- In the **Active Selectors** pane, select *Device Name*.
- Optionally, in the **Selector Definitions** pane, type an asterisk (*) in the **Device Name** field. Using the * includes all devices by Device Name. In the **Matched Devices** pane, a list of all devices appears.
- Click **[OK]** to close the modal page.
- On the **Device Group Editor** modal page, click **[Save]** and close the page.

13. Next, create a Device Group Template that will disable Event Masking for all devices in the new Device Group. Click the building blocks icon (🧱) for the new device group. A **Device Template Editor** page appears:

The screenshot displays the 'Device Template Editor' interface. At the top, there is a navigation bar with the title 'Device Template Editor | Applying Template to Devices | Click [Save] to commit changes | Config Template Settings (Click field labels to enable/dis)'. Below this, there are fields for 'Template' (set to 'New / One-off Template'), 'Save When Applied & Confirmed' (checkbox), and 'Template Name'. A tabbed interface shows 'Config' as the active tab, with other tabs including 'Interface', 'CV Policies', 'Port Policies', 'Svc Policies', 'Proc Policies', 'Dyn Apps', and 'Logs'. The main content area is divided into several sections: 'Access & Monitoring', 'Device Preferences', 'Device Retention & Basic Thresholds', and 'Interface Inventory Settings'. In the 'Access & Monitoring' section, the 'Event Mask' dropdown menu is highlighted with a red box and set to 'Disabled'. Other settings in this section include 'Device Organization' (Acme Inc), 'SNMP Read' (Cisco SNMPv2 - Example), 'SNMP Write' (None), 'Availability Protocol' (TCP), 'Avail Port' (ICMP), 'Latency Protocol' (TCP), 'Latency Port' (ICMP), 'Avail+Latency Alert' (Disabled), 'Collection' (Enabled), 'Collector Grp' (CUG), 'Coll. Type' (Standard), and 'Critical Ping' (Disabled). The 'Device Preferences' section includes options like 'Auto-Clear Events', 'Scan All IPs', 'Accept All Logs', 'Dynamic Discovery', 'Daily Port Scans', 'Preserve Hostname', 'Auto-Update', 'Disable Asset Update', and 'Bypass Interface Inventory'. The 'Device Retention & Basic Thresholds' section features sliders for 'System Latency' (100 ms), 'Daily Rollup Bandwidth Data' (730 days), 'Hourly Rollup Bandwidth Data' (120 days), 'Raw Performance Data' (7 days), 'Daily Rollup Performance Data' (730 days), 'Availability Packet Size' (56 bytes), and 'Availability Ping Count' (1 pings). The 'Interface Inventory Settings' section at the bottom shows 'Interface Inventory' (60000) and 'Maximum Allowed' (10000) with an 'Apply' button below.

14. Because all of the fields are disabled (grayed-out) by default, click the **Event Mask** field name to enable the field. Use the default setting of *Disabled*.
15. Click **[Apply]** and click **[Confirm]** on the **Device Template Editor** page.
16. Next, turn off the *Trigger on Child Rollup* option on the "ServiceNow: Add/Update Incident" Run Book Automation. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation) and click the wrench icon (🔧) for the "ServiceNow: Add/Update Incident" Run Book Automation. The **Automation Policy Editor** page appears:



17. Make sure the *Trigger on Child Rollup* option is not selected and click **[Save]**. Close the **Automation Policy Editor** page.

Hyperlinking Events

Both ServiceNow and SL1 provide mechanisms for hyperlinking to multiple active events and incidents. This section describes those processes.

ServiceNow Hyperlinking

Each incident in ServiceNow will have one or more events aligned with it through the Related List of events, which you can view in the **[Events]** tab at the bottom of the Incident record.

By default the **Hyperlink** field appears on the custom event table provided with the Certified application (`x_sclo_scilogic_event`). Each event shown in the **[Events]** tab at the bottom of the Incident record will have its own URL associated. The following image shows the Event record for an event aligned with an Incident:

Event 315

Incident: INC0010020

Region: region6341

Event ID: 315

Event policy: Dynamic App Snippet Exception

Automation policy: ServiceNow: Click to Create Incident

Action policy: ServiceNow: Add/Update/Clear Inciden

Message: em7-ao-47: App: 90, Snippet: 110 reported a collection problem (Explanation: SNMP error returned: Timeout. Location: Using walkbulk to query .1.3.6.1.2.1.25.2.3.1.2)

Device: em7-ao-47

Organization: System

Severity: Minor

Event created: 2019-10-30 13:00:19

Last detected: 2019-10-30 13:00:19

Event count: 1

Hyperlink: http://em7.mydomain.com/em7/index.em7?exec=events&q_type=aid&q_arg=315&q_sev=1&q_sort=0&q_oper=0

TIP: If the **[Events]** tab does not display at the bottom of the Incident record, you can add it by opening the Incident record, clicking the Additional actions icon (☰), and selecting *Configure > Related Lists. Add Event->Incident* to the **Selected** column and click **[Save]**.

Viewing the Incident Import Table in ServiceNow

Each time SL1 creates or changes an incident in ServiceNow, data is inserted into a temporary import table on the ServiceNow system. This table displays all inbound data from SL1 and is a useful tool to determine what data is being sent and imported. The incident import table is created automatically when you install the ScienceLogic Certified (Scoped) Application.

To view the data and the status of the import process, go to the **Import Incidents** page (ScienceLogic > Event > Events) in ServiceNow:

Event ID	Created	Correlation ID	Incident	CMDB CI	Updated	Target record	State	Incident state	Created by	Import set run
1175595	2019-09-19 09:33:21	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 09:33:21	Event_1175595	Updated	Resolved	fun	(empty)
1175595	2019-09-19 09:33:21	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 09:33:21	Incident_INC0013315	Updated	Resolved	fun	(empty)
1175595	2019-09-19 09:28:27	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 09:28:27	Event_1175595	Inserted	In Progress	fun	(empty)
1175595	2019-09-19 09:28:27	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 09:28:27	Incident_INC0013315	Updated	In Progress	fun	(empty)
1175592	2019-09-19 09:28:26	fuadem08@DEV-2197-EVENT+1705	INC0013317	(empty)	2019-09-19 09:28:26	Incident_INC0013317	Updated	Resolved	fun	(empty)
1175592	2019-09-19 09:28:26	fuadem08@DEV-2197-EVENT+1705	INC0013317	(empty)	2019-09-19 09:28:26	Event_1175592	Updated	Resolved	fun	(empty)
1175592	2019-09-19 09:23:22	fuadem08@DEV-2197-EVENT+1705	INC0013317	(empty)	2019-09-19 09:23:22	Incident_INC0013317	Updated	In Progress	fun	(empty)
1175592	2019-09-19 09:23:22	fuadem08@DEV-2197-EVENT+1705	INC0013317	(empty)	2019-09-19 09:23:22	Incident_INC0013317	Updated	In Progress	fun	(empty)
1175577	2019-09-19 09:03:34	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 09:03:34	Event_1175577	Updated	Resolved	fun	(empty)
1175577	2019-09-19 09:03:34	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 09:03:34	Incident_INC0013315	Updated	Resolved	fun	(empty)
1175577	2019-09-19 08:58:30	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:58:30	Incident_INC0013315	Updated	In Progress	fun	(empty)
1175577	2019-09-19 08:58:30	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:58:30	Event_1175577	Inserted	In Progress	fun	(empty)
1175579	2019-09-19 08:53:35	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:53:35	Incident_INC0013315	Updated	Resolved	fun	(empty)
1175579	2019-09-19 08:53:35	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:53:35	Event_1175579	Updated	Resolved	fun	(empty)
1175579	2019-09-19 08:48:31	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:48:31	Incident_INC0013315	Updated	In Progress	fun	(empty)
1175579	2019-09-19 08:48:31	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:48:31	Event_1175579	Inserted	In Progress	fun	(empty)
1175556	2019-09-19 08:38:32	fuadem08@DEV-2196-EVENT+1705	INC0013315	(empty)	2019-09-19 08:38:32	Incident_INC0013315	Updated	Resolved	fun	(empty)

You can view a complete audit of all import data and transforms by going to the **Transform Histories** page (System Import Sets > Advanced > Transform History):

Started	State	Completed	Run time	Set	Import set table	Total	Inserts	Updates	Ignored	Skipped	Errors	Transform Map
2016-09-04 04:00:50	Complete	2016-09-04 04:00:50	0 Seconds	ISE0013291	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map
2016-09-01 12:00:27	Complete	2016-09-01 12:00:27	0 Seconds	ISE0013275	ScienceLogic Network Interfaces [u_sciencelogic_adapters]	1	0	0	0	1	0	ScienceLogic Adapter T-Map
2016-09-02 14:00:12	Complete	2016-09-02 14:00:12	0 Seconds	ISE0013278	ScienceLogic Hardware Models [u_sciencelogic_hardware_models]	1	0	0	0	1	0	ScienceLogic Hardware Model T-Map
2016-09-01 16:00:44	Complete	2016-09-01 16:00:44	0 Seconds	ISE0013276	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map
2016-08-31 18:01:16	Complete	2016-08-31 18:01:16	0 Seconds	ISE0013271	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map
2016-09-02 03:00:28	Complete	2016-09-02 03:00:28	0 Seconds	ISE0013280	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map
2016-09-01 20:01:03	Complete	2016-09-01 20:01:03	0 Seconds	ISE0013276	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map
2016-09-03 02:01:22	Complete	2016-09-03 02:01:22	0 Seconds	ISE0013286	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map
2016-09-01 12:00:59	Complete	2016-09-01 12:00:59	0 Seconds	ISE0013275	ScienceLogic Network Interfaces [u_sciencelogic_adapters]	1	0	0	0	1	0	ScienceLogic Adapter T-Map
2016-09-03 12:00:12	Complete	2016-09-03 12:00:12	0 Seconds	ISE0013283	ScienceLogic Hardware Models [u_sciencelogic_hardware_models]	1	0	0	0	1	0	ScienceLogic Hardware Model T-Map
2016-09-04 22:00:33	Complete	2016-09-04 22:00:33	0 Seconds	ISE0013290	ScienceLogic Network Interfaces [u_sciencelogic_adapters]	1	0	0	0	1	0	ScienceLogic Adapter T-Map
2016-09-01 16:01:17	Complete	2016-09-01 16:01:17	0 Seconds	ISE0013276	ScienceLogic File System [u_sciencelogic_file_system]	1	0	0	0	1	0	ScienceLogic File System T-Map

ServiceNow, ScienceLogic Event, and Incident Priority Matrix

By default, when SL1 triggers an event, it is sent to ServiceNow through the Integration Service. The following mappings are currently in place for mapping an SL1 Event to a ServiceNow Incident (ScienceLogic > Event > Severity Lookup Rules):

Severity	Impact	Urgency
1	1	2
2	2	2
3	2	3
4	3	3

NOTE: This severity lookup table handles all Severity conversions.

A **transformation script** that translates the SL1 event severity into the ServiceNow **Impact**, **Urgency**, and **Priority** fields automatically deploys with the ScienceLogic Certified (Scoped) Application.

By default, the **Priority** field is read-only and must be set by selecting the **Impact** and **Urgency** values.

Priority is calculated according to the following data lookup rules:

Impact	Urgency	Priority
1 - High	1 - High	1 - Critical
1 - High	2 - Medium	2 - High
1 - High	3 - Low	3 - Moderate
2 - Medium	1 - High	2 - High
2 - Medium	2 - Medium	3 - Moderate
2 - Medium	3 - Low	4 - Low
3 - Low	1 - High	3 - Moderate
3 - Low	2 - Medium	4 - Low
3 - Low	3 - Low	5 - Planning

Adding Additional Fields to the Transform Map

If you require additional mandatory fields to be filled out to resolve an incident, you can add those fields to the **transform map** in ServiceNow.

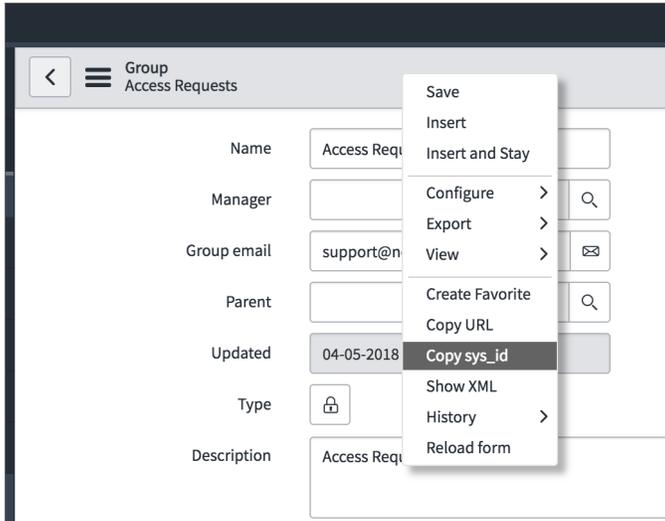
For example, if you require four mandatory fields in the ServiceNow Incident—**Assignment Group**, **IT Service**, **Service Component**, and **Description**—to be filled out before that incident can be resolved in SL1, you would perform the following steps.

The screenshot displays the 'Main Details' section of a ServiceNow Incident form. The form is organized into two columns of fields. The left column includes fields for Caller (Science Logic), Behalf of user, Company (Motorpoint Limited), Location, IT service, Service component, Symptom, Configuration item, and Short description. The right column includes fields for Contact type (web), Impact (Some Users), Urgency (3 - Low), Priority (4 - Low), Major incident state, Owing group, Assignment group, and Assigned to. A 'Related Search Results' section is located below the main form fields. The fields for IT service, Service component, Assignment group, and Description are highlighted with red rectangular boxes, indicating they are mandatory for resolution.

Field Name	Value
Caller	Science Logic
Behalf of user	
Company	Motorpoint Limited
Location	
IT service	
Service component	
Symptom	
Configuration item	
Short description	mp-sql-2014-01: Host Resource: Storage Utilization (E:\Label\Data Serial Number d01ef7f2) has exceeded threshold 90%, currently 90.02%
Contact type	web
Impact	Some Users
Urgency	3 - Low
Priority	4 - Low
Major incident state	
Owing group	
Assignment group	
Assigned to	

To add an assignment group:

1. Navigate to **User Administration > Groups** and select the assignment group you want to add. The Group record appears.
2. Right-click the gray task bar at the top and select **Copy sys_id**.



3. In SL1, open to the "ServiceNow: Add/Update/Clear Incident" Run Book Action (Registry > Run Book > Actions).
4. Edit the **Input Parameters** of the Run Book Action to add the **sys_id** to the relevant parameter or parameters to assign the assignment group to one of the new, acknowledged, or cleared incidents that are mapped. After an incident is created, the assignment group value will not be changed by the Run Book Action.

In the following example, the assignment group is assigned to incidents that are *cleared*:

```
"assignment_group_new": "",  
"assignment_group_ack": "",  
"assignment_group_clear": "sys_id"
```

The **IT Service**, **Service Component**, and **Description** fields in our example must be filled in before an Incident can be closed. To do this, changes must be made in the transform maps that are provided in the form of update sets from ScienceLogic.

TIP: For more information about mapping new fields and other mappings options, see https://docs.servicenow.com/bundle/newyork-platform-administration/page/script/server-scripting/concept/c_MappingOptions.html.

To add the **Description** field:

1. In your ServiceNow instance, search for "transform map" in the left-hand menu. Click **Transform Maps**.
2. In the list of transform maps, search for "ScienceLogic" in the field above the **Name** column.

- Open the "ScienceLogic Incident" map:

	Name	Source table	Target table	Run business rules	Order	Active	Updated
<input type="checkbox"/>	ScienceLogic Incident	ScienceLogic Incident Import [u_imp_silo_incidents]	Incident [incident]	true	100	true	2018-05-18 20:44:15
<input type="checkbox"/>	Notification	Notification [imp_notification]	Incident [incident]	true	100	true	2008-12-28 22:08:39
<input type="checkbox"/>	User	User [imp_user]	User [sys_user]	true	100	true	2015-07-07 07:32:42
<input type="checkbox"/>	Location	Location [imp_location]	Location [cmn_location]	true	100	true	2015-07-07 07:30:52
<input type="checkbox"/>	Computer	Computer [imp_computer]	Computer [cmdb_ci_computer]	true	100	true	2015-07-07 07:29:19
<input type="checkbox"/>	LDAP User Import	label [ldap_import]	User [sys_user]	true	100	true	2011-04-11 16:17:55
<input type="checkbox"/>	LDAP Group Import	label [ldap_group_import]	Group [sys_user_group]	true	100	true	2009-11-12 09:33:59
<input type="checkbox"/>	ScienceLogic Event	ScienceLogic Incident Import [u_imp_silo_incidents]	ScienceLogic Events [u_silo_events]	true	200	true	2018-05-18 21:27:35

- The **Field Maps** table at the bottom of the page allows you to edit or create mappings from the ScienceLogic Incident Import table to the ServiceNow Incident table. Click **[New]** to create a new field mapping.
- The **Source table** field should contain the ScienceLogic Incident Import and the **Target table** should include the ServiceNow Incident table:

Field Map
New record

Map: ScienceLogic Incident

Source table: ScienceLogic Incident Import [u_imp_s...]

Source field: Action Policy

Use source script:

Application: Global

Target table: Incident [incident]

Target field: Active

Coalesce:

- To create a new mapping to copy the contents of the **Short description** field to the **Description** field, select **Short description** from the **Source field** drop-down menu.
- In the **Target field** drop-down menu, select **Description**.
- Click **Update** to save your changes.

The **IT Service** and **Service Component** fields in our example are set in the Transform Script in the "ScienceLogic Event" transform map. To set the fields:

- Make sure you have the `sys_id` for the target fields. These can be found in ServiceNow. If a field contains a magnifying glass, it will require a `sys_id`. If a field has a drop-down, then type in the field you wish to apply from the drop-down. In the case of our example, the `sys_ids` of the two fields are required.
- In your ServiceNow instance, navigate to the **Transform Maps** table and select "ScienceLogic Event".

3. In the ScienceLogic Event transform map page, click the **Transform Script** tab and open the "onAfter" script.

Table Transform Map
ScienceLogic Event

Current View: Default view (click me for other views)

* Source table: ScienceLogic Incident Import [u_imp_s...]

Active:

Run business rules:

Enforce mandatory fields: No

Copy empty fields:

Update Copy Delete

Application: ScienceLogic

Created: 2018-05-18 21:27:35

* Target table: ScienceLogic Events [u_silo_events]

Order: 200

Run script:

Related Links

- [Auto Map Matching Fields](#)
- [Transform](#)
- [Index Coalesce Fields](#)

Field Maps (16) Transform Scripts (2)

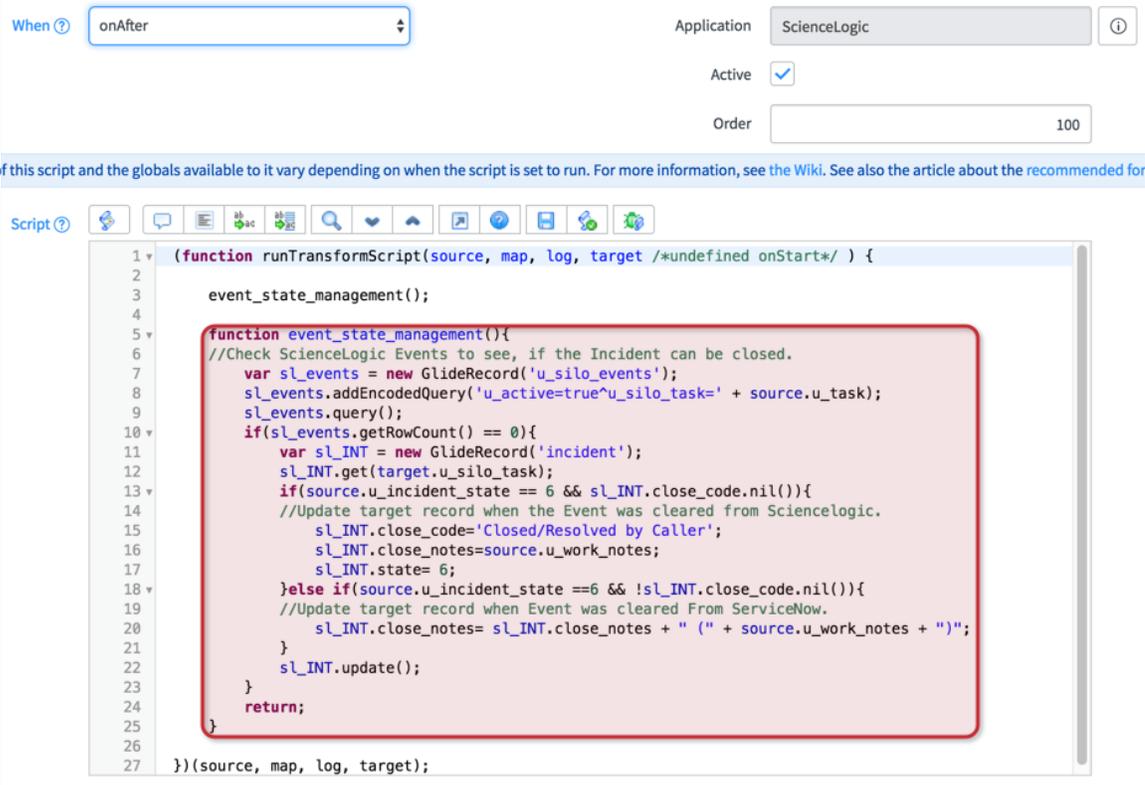
Transform Scripts New Go to Order Search

Map = ScienceLogic Event

When	Script	Order	Active
onAfter	(function runTransformScript(source, map...	100	true
onBefore	(function runTransformScript(source, map...	100	true

4. Add the following under the "//Update target record when the Event was cleared from Sciencelogic" text:

```
sl_INT.(target field) = '[sys_id of the source field]'; //(IT service field)
sl_INT.(target field) = '[sys_id of the source field]'; //(Service component)
```



The screenshot shows a script editor interface for ScienceLogic. At the top, the "When" dropdown is set to "onAfter", the "Application" is "ScienceLogic", "Active" is checked, and "Order" is 100. Below the header, there is a toolbar with various icons. The main area contains a JavaScript function named "runTransformScript" which calls "event_state_management". The "event_state_management" function is highlighted with a red border and contains the following code:

```
1 (function runTransformScript(source, map, log, target /*undefined onStart*/ ) {
2
3     event_state_management();
4
5     function event_state_management(){
6         //Check ScienceLogic Events to see, if the Incident can be closed.
7         var sl_events = new GlideRecord('u_silo_events');
8         sl_events.addEncodedQuery('u_active=true^u_silo_task=' + source.u_task);
9         sl_events.query();
10        if(sl_events.getRowCount() == 0){
11            var sl_INT = new GlideRecord('incident');
12            sl_INT.get(target.u_silo_task);
13            if(source.u_incident_state == 6 && sl_INT.close_code.nil()){
14                //Update target record when the Event was cleared from ScienceLogic.
15                sl_INT.close_code='Closed/Resolved by Caller';
16                sl_INT.close_notes=source.u_work_notes;
17                sl_INT.state= 6;
18            }else if(source.u_incident_state ==6 && !sl_INT.close_code.nil()){
19                //Update target record when Event was cleared From ServiceNow.
20                sl_INT.close_notes= sl_INT.close_notes + " (" + source.u_work_notes + ")";
21            }
22            sl_INT.update();
23        }
24        return;
25    }
26
27 }(source, map, log, target);
```

- To find the target field, make a temporary mapping to see what the target field is. This mapping can be deleted once you know the target field.

The screenshot shows the 'Field Maps' interface with a table of mappings. The table has two columns: 'Source field' and 'Target field'. The 'u_work_notes' source field is mapped to the 'u_service_component' target field. A red arrow points from the 'u_work_notes' source field to the 'u_service_component' target field, which is highlighted with a red box.

	Source field	Target field
<input type="checkbox"/>	u_short_description	shprt_description
<input type="checkbox"/>	u_contact_type	contact_type
<input type="checkbox"/>	u_active	active
<input type="checkbox"/>	u_short_description	description
<input type="checkbox"/>	u_assignment_group	assignment_group
<input type="checkbox"/>	u_correlation_id	correlation_id
<input type="checkbox"/>	sys_updated_by	caller_id
<input type="checkbox"/>	u_urgency	urgency
<input type="checkbox"/>	u_cmdb_ci	cmdb_ci
<input type="checkbox"/>	u_work_notes	work_notes
<input checked="" type="checkbox"/>	u_impact	u_service_component
<input type="checkbox"/>	u_impact	impact
<input type="checkbox"/>	[Script]	company
<input type="checkbox"/>	[Script]	location

- Click **[Update]** to save your changes. The selected fields will be added into an Incident on closure.

Chapter

4

ServiceNow CMDB SyncPack

Overview

This chapter describes the *ServiceNow CMDB SyncPack*, which is the ScienceLogic integration with the ServiceNow Configuration Management Database (CMDB).

This SyncPack maintains and enhances the ServiceNow CMDB by sharing discovered device information, importing and exporting data bi-directionally between SL1 and ServiceNow, and by automatically maintaining ServiceNow Configuration Item (CI) relationships.

This chapter covers the following topics:

<i>Downloading, Importing, and Installing the SyncPack</i>	72
<i>Integration Applications Included in the SyncPack</i>	74
<i>Installing Additional Components Used by the SyncPack</i>	76
<i>Syncing Organizations</i>	91
<i>Syncing Devices from SL1 to ServiceNow</i>	93
<i>Syncing CI Attributes from ServiceNow to SL1</i>	105
<i>Syncing Advanced Topology Data from SL1 to ServiceNow</i>	107
<i>Syncing Network Interfaces from SL1 to ServiceNow</i>	109
<i>Syncing File Systems from SL1 to ServiceNow</i>	111
<i>Syncing Business Services from SL1 to ServiceNow</i>	113
<i>Syncing Installed Software between SL1 and ServiceNow</i>	115
<i>Discovery Sync</i>	116
<i>Scheduling Integration Applications</i>	131
<i>Troubleshooting CMDB Sync</i>	134

Downloading, Importing, and Installing the SyncPack

A SyncPack file has the **.whl** file extension type. You can download the SyncPack file from the ScienceLogic Support site.

Downloading the SyncPack

To locate and download the SyncPack:

1. Go to the ScienceLogic Support site at <https://support.sciencelogic.com/s/>.
2. Click the **Product Downloads** tab, select *PowerPacks*, and then click the "Synchronization" link. The **Synchronization PowerPack Downloads** page appears.
3. Select the SyncPack you want to install.
4. In the **Files** list, select the SyncPack **.whl** file and click **[Download]**.

NOTE: SyncPacks do not require a specific license. After you download a SyncPack, you can import it to your Integration Service using the Integration Service user interface.

Importing the SyncPack

NOTE: You must import and install the *ServiceNow Base SyncPack* before uploading and installing any of the other ServiceNow SyncPacks.

To import a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click **[Import SyncPack]**. The **Import SyncPack** page appears.
2. Click **[Browse]** and select the **.whl** file for the SyncPack you want to install.

TIP: You can also drag and drop a **.whl** file to the **SyncPacks** page.

3. Click **[Import]**. The Integration Service registers and uploads the SyncPack. The SyncPack is added to the **SyncPacks** page.

NOTE: You cannot edit the content package in a SyncPack published by ScienceLogic. You must make a copy of a ScienceLogic SyncPack and save your changes to the new SyncPack to prevent overwriting any information in the original SyncPack when upgrading.

Installing the SyncPack

To install a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click the **[Actions]** button () for the SyncPack you want to install and select *Activate & Install*. The **Activate & Install SyncPack** modal appears.

TIP: By default, the **SyncPacks** page only displays activated and installed SyncPacks. If you do not see the SyncPack that you want to install, click the toggle icon () on the **SyncPacks** page and select *Show All SyncPacks* to see a list of the uninstalled SyncPacks.

2. Click **[Yes]** to confirm the activation and installation. When the SyncPack is activated, the **SyncPacks** page displays a green check mark icon () for that SyncPack. If the activation or installation failed, then a red exclamation mark icon () appears.
3. For more information about the activation and installation process, click the check mark icon () or the exclamation mark icon () in the **Activated** column for that SyncPack. For a successful installation, the "Activate & Install SyncPack" integration application appears, and you can view the Step Log for the steps. For a failed installation, the **Error Logs** window appears.

Integration Applications Included in the SyncPack

This section lists the integration applications that are in the ServiceNow CMDB SyncPack.

Integration Applications

The following integration applications are included with the ServiceNow SyncPack:

- **Cache ServiceNow CIs and SL1 Device Classes.** Reads all existing SL1 Device Classes and ServiceNow CIs and writes them to a cache. To perform a Device Sync, run this integration application before you run the "Sync Devices from SL1 to ServiceNow" integration application. For more information, see [Syncing Devices from SL1 to ServiceNow](#).
- **Delete Devices from SL1.** Lets you delete devices in a specific SL1 Virtual Collector Group (VCUG) if those devices have not been modified in SL1 for a specified amount of time that is set in the application.
- **Generate Required CI Relations for ServiceNow.** Pulls device class mappings from the "Sync Devices from SL1 to ServiceNow" and the "Sync CI Attributes from ServiceNow to SL1" integration applications to prevent you from having to add a separate set of class mappings. The application also lists any missing relationships in the Step Log in the Integration Service user interface. For more information about the log messages, see [Log Messages for the "Generate Required CI Relations for ServiceNow" Application Integration](#).
- **Sync Advanced Topology from SL1 to ServiceNow.** Reads Dynamic Component Mapping relationships from SL1 and syncs those relationships with ServiceNow. You must run both the "Sync Devices from SL1 to ServiceNow" application and the "Sync Interfaces from SL1 to ServiceNow" application at least twice on new Integration Service systems to populate the cache for this integration application. For more information, see [Syncing Advanced Topologies from SL1 to ServiceNow](#).
- **Sync Business Services from SL1 to ServiceNow.** Reads Business Services, IT Services, and Device Services in SL1 and syncs them with business services in ServiceNow. This integration application creates and updates services, but it does not delete services. For more information, see [Syncing Business Services from SL1 to ServiceNow](#).
- **Sync CI Attributes from ServiceNow to SL1.** Reads CI attributes from ServiceNow and maps those attributes to asset and attribute fields in SL1. This application uses the mappings and additional attributes options from the "Sync Devices from SL1 to ServiceNow" application. This integration application can also sync the location and production state attributes from ServiceNow to SL1. For more information, see [Syncing CI Attributes from ServiceNow to SL1](#).
- **Sync Device Groups from SL1 to ServiceNow.** Collects all device groups and group IDs from SL1 and posts device group data to ServiceNow. To prevent errors when running this application or a device sync, make sure that the device group names are not already being used by existing groups in ServiceNow. For more information about Device Sync, see [Syncing Devices from SL1 to ServiceNow](#).
- **Sync Devices from SL1 to ServiceNow.** Syncs devices and their properties and relationships from SL1 to ServiceNow. For more information, see [Syncing Devices from SL1 to ServiceNow](#).
- **Sync Discovery Requirements.** Processes credentials from SL1, processes collector groups, device templates, virtual device classes, and collectors, and then syncs organizations and device groups. For more information, see [Discovery Sync](#).

- **Sync Discovery Session Status from SL1 to ServiceNow.** Collects and processes Discovery sessions from SL1, and collects Discovery session logs. For more information, see [Discovery Sync](#).
- **Sync File Systems from SL1 to ServiceNow.** Reads file systems discovered in SL1 and then maps them to a parent CI record in ServiceNow. For more information, see [Syncing File Systems from SL1 to ServiceNow](#).
- **Sync Installed Software from SL1 to ServiceNow.** Reads all available software packages from ServiceNow and the devices aligned to that software by region and syncs them with SL1. For more information, see [Syncing Installed Software between SL1 and ServiceNow](#).
- **Sync Interfaces from SL1 to ServiceNow.** Collects network interface data from ServiceNow and SL1, and then runs multiple CI syncs for each interface to be synced. For more information, see [Syncing Network Interfaces from SL1 to ServiceNow](#).
- **Sync Organizations from SL1 to ServiceNow.** Pulls organizations from SL1 and syncs to ServiceNow. For more information, see [Syncing Organizations from SL1 to ServiceNow](#).
- **Sync Service Requests from ServiceNow to SL1.** Processes Discovery sessions and posts Discovery sessions and new virtual devices to SL1. Also enables device decommissioning for devices you no longer want to monitor. This application was formerly named "Sync Discovery Session Requests from ServiceNow to SL1".
- **Sync Software Packages from SL1 to ServiceNow.** Reads all software packages from and creates new CIs in ServiceNow. Run this integration before running the "Sync Installed Software" integration application. For more information, see [Syncing Installed Software between SL1 and ServiceNow](#).

Integration Applications (Internal)

To view the internal integration applications, click the Filter icon () on the **Integrations** page and select *Show Hidden Integrations*. Internal integration applications are hidden by default. The following integration applications are "internal" applications that should not be run directly, but are automatically run by applications from the previous list:

- **Bulk Delete Devices.** Deletes devices from SL1.
- **Create Discovery Session in SL1.** Creates and starts a Discovery session in SL1 and updates the ServiceNow service request.
- **Create ServiceNow CI.** Creates a new ServiceNow CI with a mappings dictionary, but does not attempt to look up new CIs.
- **Create Virtual Device in SL1.** Creates a virtual device in SL1 and updates the Requested Item (RITM) value.
- **Post Attribute DB Calls to SL1.** Posts attribute database calls to SL1.
- **Post Attribute Rest Calls to SL1.** Posts attribute REST calls to SL1.
- **Post Company and Organization Updates.** Posts company and organization updates to ServiceNow or SL1.
- **Post Discovery-dependent Data to ServiceNow.** Posts data used by a Discovery session to ServiceNow.
- **Post Installed Software to ServiceNow.** Posts installed software data to ServiceNow.
- **Post New Companies to ServiceNow.** Posts new companies to ServiceNow.
- **Post New Organization to SL1.** Posts a new organization to SL1.

- **Process Remove Device Requests from ServiceNow to SL1.** Pulls requested device information from SL1 and validates the requests to remove a device from monitoring. Removed devices are placed in an SL1 Virtual Collector Group.
- **Pull and Post Discovery Logs.** Pulls Discovery session logs from SL1 and posts updates to ServiceNow.

Installing Additional Components Used by the SyncPack

Before you can use the ServiceNow CMDB SyncPack, you must complete the following installation and configuration steps:

1. [Install the ScienceLogic SL1: CMDB & Incident Automation application in ServiceNow](#)
2. [Install and activate the CMDB plugin in ServiceNow](#)
3. [Enable the ServiceNow Identification and Reconciliation Module](#)
4. For domain-separated ServiceNow instances only, [install the ScienceLogic Domain Separation \(Global\) update set in ServiceNow](#)
5. [Create a ServiceNow update set with containment rules and hosting rules for Device Sync](#)

Installing the ScienceLogic SL1: CMDB & Incident Automation Application in ServiceNow

You must install the "ScienceLogic SL1: CMDB & Incident Automation" application on your ServiceNow instance to enable this SyncPack. The "ScienceLogic SL1: CMDB & Incident Automation" application is also known as the "Certified Application" or the "Scoped Application".

NOTE: You must have a ServiceNow HI Service Account to request this application and download it onto your ServiceNow instance.

WARNING: Integration Service instances running version 2.0.0 or later of the ServiceNow integration applications are not backwards-compatible with the previous ServiceNow update sets or with SyncServer. After you install the "ScienceLogic SL1: CMDB & Incident Automation" application on your ServiceNow instance, you need to upgrade your ServiceNow integration applications to version 2.0.0 or later on all Integration Service instances. The "ScienceLogic SL1: CMDB & Incident Automation" application is also not backwards-compatible with SyncServer. This change cannot be reverted.

You must first request the "ScienceLogic SL1: CMDB & Incident Automation" application from the ServiceNow Store, and then you can install it.

To request and install the Certified Application:

1. Go to the ServiceNow Store at <https://store.servicenow.com> and search for "ScienceLogic SL1".
2. Select the "ScienceLogic SL1: CMDB & Incident Automation" application. The detail page for the application appears.
3. Click the **[Get]** button and log in with your HI credentials.
4. After the request is approved, log in to ServiceNow as an administrator and navigate to **Application Manager** (System Applications > Applications).
5. Click **[Downloads]** in the menu header or search for "ScienceLogic".
6. Click the version drop-down for the "ScienceLogic ServiceNow Integration" application listing to make sure you are using the correct version of the application that is compatible with your version of this SyncPack.
7. Click the **[Install]** button for the "ScienceLogic ServiceNow Integration" application. The installation is complete when the button changes to **[Installed]**.
8. In the filter navigator, search for "ScienceLogic" and locate the application in the left-hand navigation menu.

NOTE: You might need to individually upload the "Sync Devices from SL1 to ServiceNow" integration application a second time after the bulk upload to ensure that the application picks up the correct application variable formatting in the user interface.

Installing and Activating the CMDB Plugin in ServiceNow

Installing the ServiceNow Configuration Management for Scoped Apps (CMDB) Plugin is required to manage your Configuration Items. This involves activating the Configuration Management For Scoped Apps (CMDB) Plugin on your ServiceNow instance.

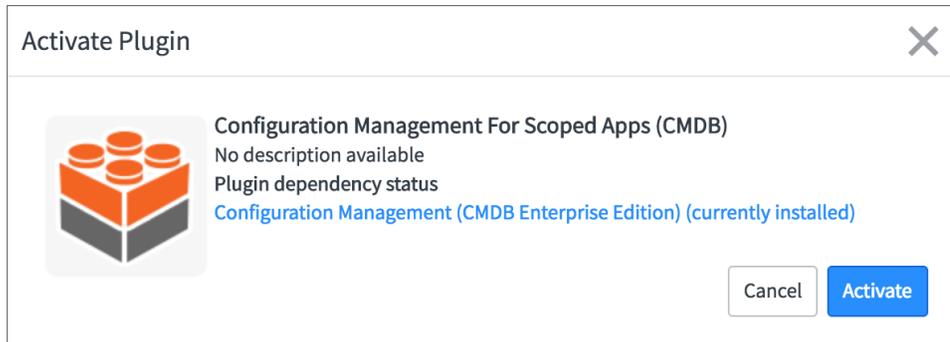
To activate the Configuration Management for Scoped Apps (CMDB) plugin:

1. Log in to ServiceNow as an administrator, and then navigate to **Plugins** (System Definition > Plugins).
2. Search for **Configuration Management For Scoped Apps (CMDB)** and click on it.



	*configuration Management	Search	Search	Search
<input type="checkbox"/>	Configuration Management (CMDB Enterprise Edition)	1.0	Active	com.snc.cmdb.enterprise
<input type="checkbox"/>	Configuration Management (CMDB)	1.1	Active	com.snc.cmdb
<input type="checkbox"/>	Configuration Management For Scoped Apps (CMDB)	1.0.0	Inactive	com.snc.cmdb.scoped
<input type="checkbox"/>	Performance Analytics - Content Pack - Configuration Management (CMDB)	1.0.0	Inactive	com.snc.pa.cmdb

3. Click *Activate/Upgrade* in the **Related Links** section.
4. In the Activate Plugin notification, click **[Activate]**.



Enabling the ServiceNow Identification and Reconciliation Module

The ServiceNow SyncPack uses the ServiceNow Identification and Reconciliation module to create and de-duplicate CI records. The Integration Service builds a JSON-formatted string that is sent to the ServiceNow Identification and Reconciliation module. The following link provides additional detail about the formatting of the JSON-formatted string: [IdentificationEngineScriptableApi](#).

The JSON-formatted string is sent directly to a custom-scripted API endpoint and run through the IdentificationEngineScriptable API. Identification (Insert or Update) of Configuration Items (CIs) is handled by the ServiceNow Identification and Reconciliation module.

For more information about the Identification and Reconciliation module, see [CMDB Identify and Reconcile](#). See also [Reconciliation Rules](#), [CMDB Identification Rules](#), and [Identification engine error messages](#).

Configuring Service Rules for Device Sync

The ServiceNow SyncPack utilizes class hierarchies to build relationships in ServiceNow. This requires building **service rules** (containment rules and hosting rules) in ServiceNow to correctly identify dependent CIs during the business discovery process and service mapping. **Containment rules** describe which CIs are contained by a given CI. **Hosting rules** describe the environment on which a CI runs.

ScienceLogic recommends packaging all of the service rules into a ServiceNow update set so that you can be easily package and deploy these changes across environments. For more information, see [Creating a ServiceNow Update Set](#).

These rules or "mappings" are defined in the "Sync Devices from SL1 to ServiceNow" integration application in the Integration Service user interface. These mappings connect an **SL1 device class** to a **ServiceNow CI class**, which determines the CI class that ServiceNow uses when creating the CI in ServiceNow.

For more information about the Identification and Reconciliation module, see [CMDB dependent relationship rules](#) and [CMDB Identification Rules](#) at the ServiceNow website.

For example, if you experience error messages about missing relationships in ServiceNow when you run the "Sync Devices from SL1 to ServiceNow" integration application in the Integration Service user interface, you might be missing certain containment rules or mappings that are needed to complete the export process:

MODULE	DATE/TIME	LOG LEVEL	MESSAGE
1 ipaas_logger	2018-08-08 17:41:29,245	FLOW	Start SnowEM7DeviceComparator
2 BaseStep	2018-08-08 17:41:29,274	ERROR	Traceback (most recent call last): File "/usr/local/lib/python2.7/site-packages/ipaascore/BaseStep.py", line 458, in execute_step self.execute() File "SnowEM7DeviceCompare", line 70, in execute File "SnowEM7DeviceCompare", line 191, in get_device_dicts File "SnowEM7DeviceCompare", line 149, in get_parent_device_info StepFailedException: No ci match in cache for Parent/Child Relationship: cmdb_ci_netapp_datacenter . Please verify the existence of the CIs, and their relationship in ServiceNow

Containment Rules

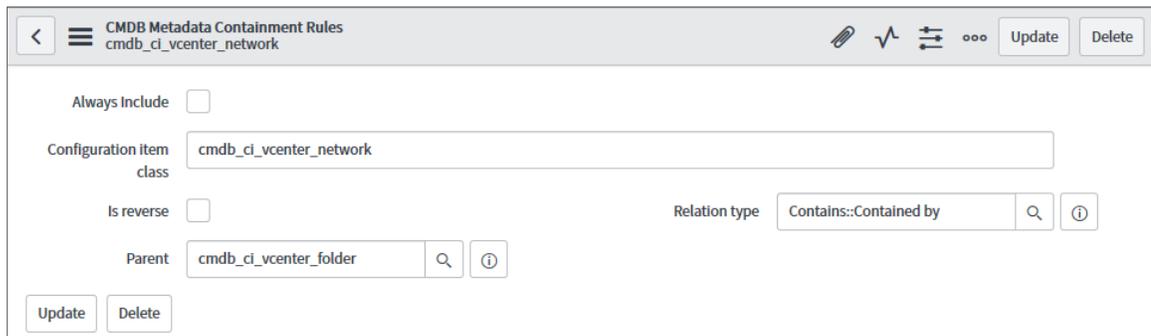
Containment rules are chained to each other in a group, with a CI type that is the top-level (root) parent of the group.

To create containment rules:

1. Log in to your ServiceNow instance.
2. In the filter navigator, type "cmdb_metadata_containment.list" to view the **CMDB Metadata Containment Rules** page:

	Always Include	Configuration item class	Is reverse	Parent	Relation type
<input type="checkbox"/>	false	cmdb_ci_esx_resource_pool	false	cmdb_ci_esx_resource_pool	Contains-Contained by
<input type="checkbox"/>	false	cmdb_ci_vcenter_datastore	false	cmdb_ci_vcenter_folder	Contains-Contained by
<input type="checkbox"/>	false	cmdb_ci_linux_server	false	cmdb_ci_vcenter_cluster	Hosted on-Hosts
<input type="checkbox"/>	false	cmdb_ci_vmware_instance	false	cmdb_ci_vcenter_cluster	Contains-Contained by
<input type="checkbox"/>	false	cmdb_ci_esx_server	false	cmdb_ci_vcenter_cluster	Contains-Contained by

3. Click **[New]**. A new CMDB Metadata Containment Rules record appears:



The screenshot shows a web form titled "CMDB Metadata Containment Rules" with the breadcrumb "cmdb_ci_vcenter_network". The form includes the following fields and controls:

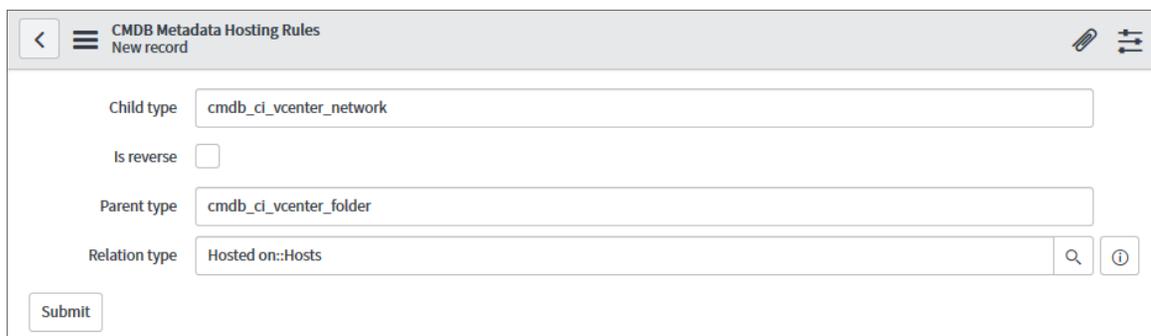
- Always Include:** A checkbox that is currently unchecked.
- Configuration item class:** A text input field containing "cmdb_ci_vcenter_network".
- Is reverse:** A checkbox that is currently unchecked.
- Relation type:** A dropdown menu showing "Contains::Contained by".
- Parent:** A text input field containing "cmdb_ci_vcenter_folder" with search and help icons.
- Buttons:** "Update" and "Delete" buttons are located at the bottom left.

4. In the New Metadata Containment Rules record, complete the following fields:
 - **Configuration item class.** Specify the child CI class.
 - **Parent.** Specify the parent CI class.
 - **Relation type.** Specify the relationship type. The common relationship types used by the ServiceNow integration are "contained" or "contained by", depending on your CMDB. Click the magnifying glass icon to select the correct value.
5. Click **[Submit]**.
6. In the Integration Service user interface, go to the **Integrations** page and manually run the "Cache ServiceNow CIs and SL1 Device Classes" integration application.
7. Run the "Sync Devices from SL1 to ServiceNow" integration application and make sure that no errors exist due to missing CI relationships.

Hosting Rules

Hosting rules can only be one level, and they always involve resources such as physical or virtual hardware.

1. In the ServiceNow filter navigator, type "cmdb_metadata_hosting.list" to view the **CMDB Metadata Hosting Rules** page.
2. Click **[New]**. A new CMDB Metadata Hosting Rules record appears:



The screenshot shows a web form titled "CMDB Metadata Hosting Rules" with the breadcrumb "New record". The form includes the following fields and controls:

- Child type:** A text input field containing "cmdb_ci_vcenter_network".
- Is reverse:** A checkbox that is currently unchecked.
- Parent type:** A text input field containing "cmdb_ci_vcenter_folder".
- Relation type:** A dropdown menu showing "Hosted on::Hosts".
- Buttons:** A "Submit" button is located at the bottom left.

3. In the New Metadata Hosting Rules record, complete the following fields:

- **Child type.** Specify the child CI class.
 - **Parent type.** Specify the parent CI class.
 - **Relation type.** Specify the relationship type. The common relationship types used by the ServiceNow integration are "Hosts" or "Hosted on", depending on your CMDB. Click the magnifying glass icon to select the correct value.
4. Click **[Submit]**.
 5. Add any additional containment and hosting rules that are needed to build the CI relationships in ServiceNow.
 6. In the Integration Service user interface, go to the **Integrations** page and manually run the "Cache ServiceNow CIs and SL1 Device Classes" integration application.
 7. Run the "Sync Devices from SL1 to ServiceNow" integration application and make sure that no errors exist due to missing CI relationships.

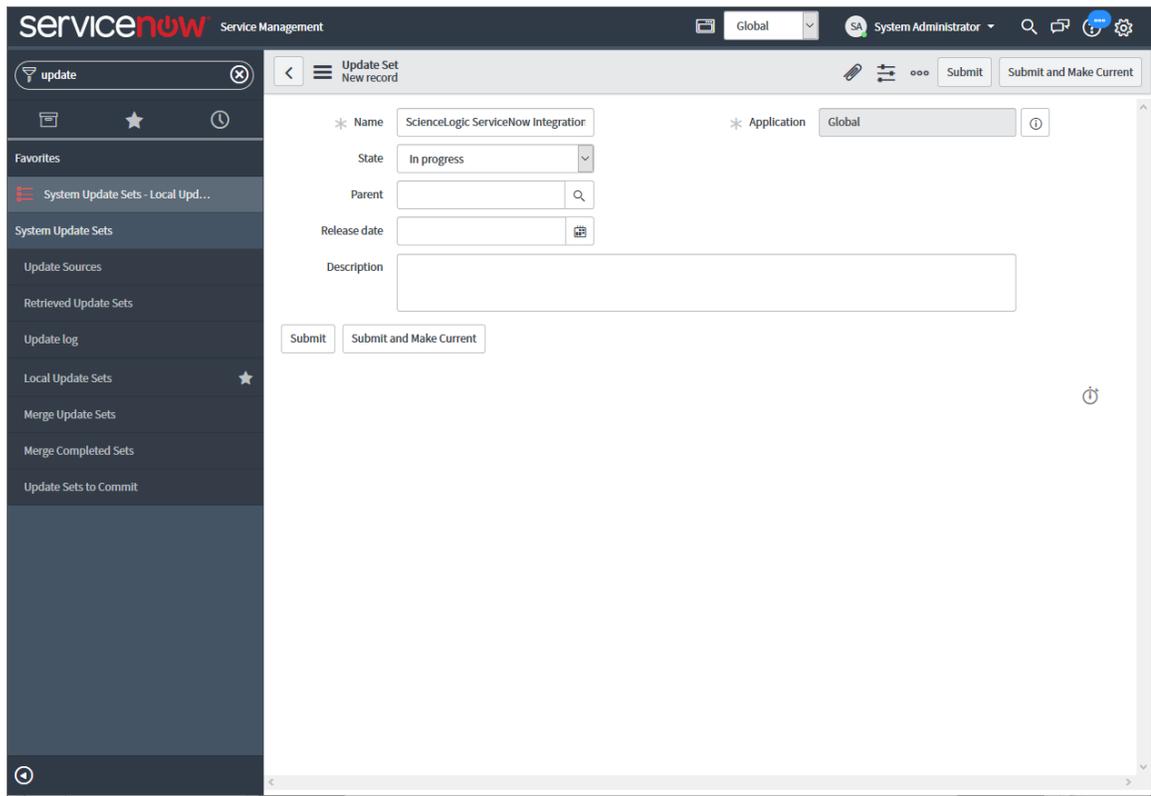
Creating a ServiceNow Update Set

ScienceLogic recommends packaging the service rules into a standalone ServiceNow update set that you can export if needed. An **update set** is an XML file containing a group of customizations that can be moved from one ServiceNow instance to another. This update set should include any changes or configurations to the service rules for the ServiceNow Identification and Reconciliation Module.

To create a standalone update set in ServiceNow:

1. Log in to your ServiceNow instance.
2. Enable the Developer Update set picker by clicking the **Settings** icon () and selecting the **Developer** tab.
3. Select the **Show update set picker in header** toggle to enable it, and then close the **System Settings** page.
4. In the filter navigator, search for local update sets.

5. Under **System Update Sets**, select **Local Update Sets** and click **[New]**. A new Update Set record appears:



The screenshot shows the ServiceNow interface for creating a new Update Set. The left sidebar is open to 'Local Update Sets'. The main form area is titled 'Update Set New record' and contains the following fields:

- Name:** ScienceLogic ServiceNow Integrator
- Application:** Global
- State:** In progress
- Parent:** (empty field with search icon)
- Release date:** (empty field with calendar icon)
- Description:** (empty text area)

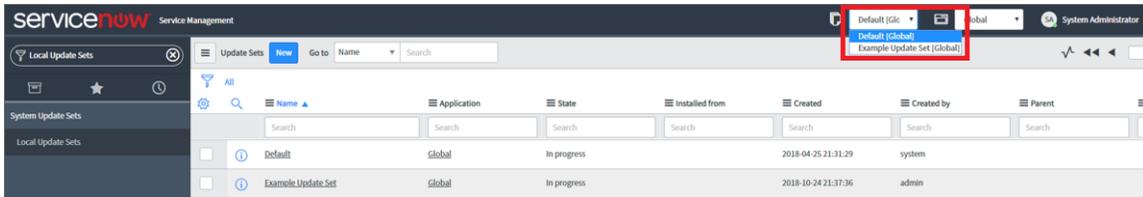
At the bottom of the form are two buttons: 'Submit' and 'Submit and Make Current'.

6. Complete the following fields:
 - **Name.** Specify a name that describes the rules of this update set.
 - **Application.** Set the application scope to *Global*.
 - **State.** Set to *In Progress*.
 - Complete the remaining fields as needed.
7. Click **[Submit]** or **[Submit and Make Current]**. If you selected **[Submit and Make Current]**, go to step 9.
8. If you clicked **[Submit]**, you can select the update set in the picker in the header or navigate to the update set and select *Make This My Current Set* in the **Related links** section.
9. You are ready to make changes in your ServiceNow Instances.
10. When you are done with all updates in the update set, change the update set **State** field to *Complete*.

Adding Service Rules to an Update Set

If you submitted your new update set and made it "Current" in [Creating a ServiceNow Update Set](#), skip this step and go to [Exporting an Update Set](#).

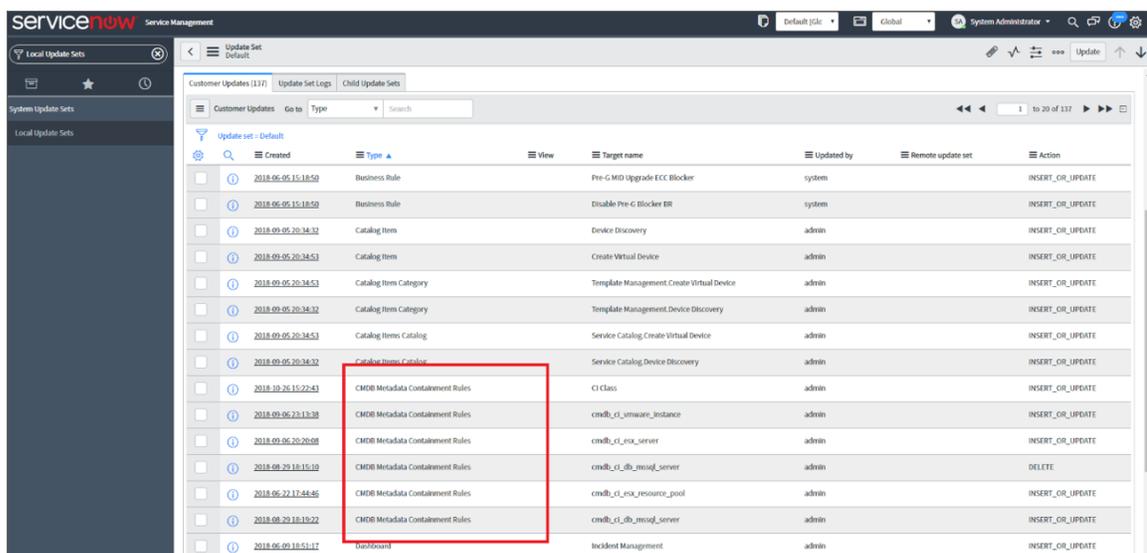
If you did not make your update set current, you will need to identify your current update set and move all of the service rules you need into your update set. You can find this information in a dropdown located in the ServiceNow navigation bar:



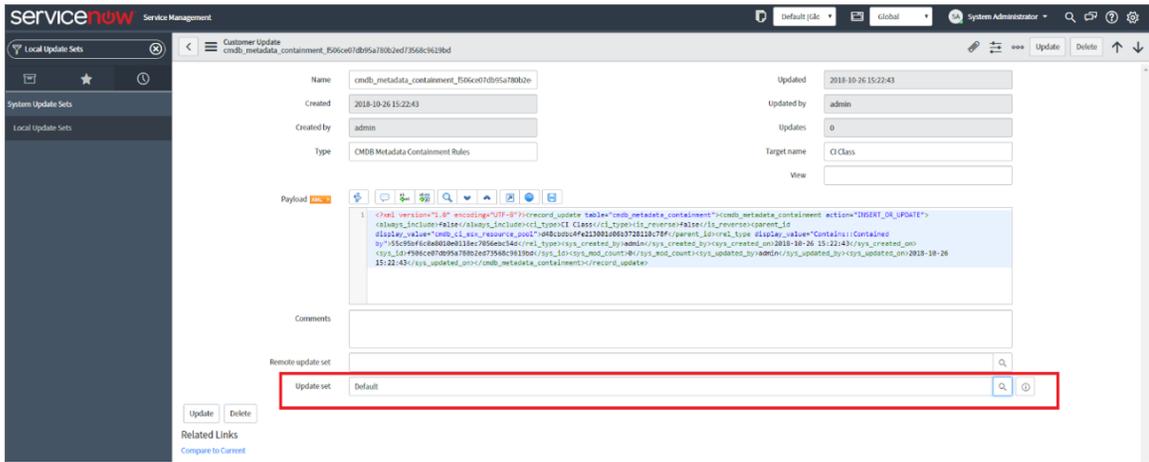
All of the service rules that you defined are tracked in the update set record under the **[Customer Updates]** tab.

To add all created service rules to your update set:

1. Log in to your ServiceNow instance.
2. In the filter navigator, type "local update sets" to view a list of update sets on the ServiceNow instance.
3. Identify your current update set, which should have all of the created service rules tracked.
4. Identify the self-created update set that you want to contain all the service rules. This is the update set that you want to export.
5. Select the current update set that has all of the already-created service rules.
6. On the **[Customer Updates]** tab, identify all of the records with a Type of either *CMDB Metadata Containment Rules* or *CMDB Metadata Hosting Rules*:



7. Select each of the relevant service rule records and set the **Update set** field to match the update set you want to export. Click the magnifying glass icon to select the correct value.



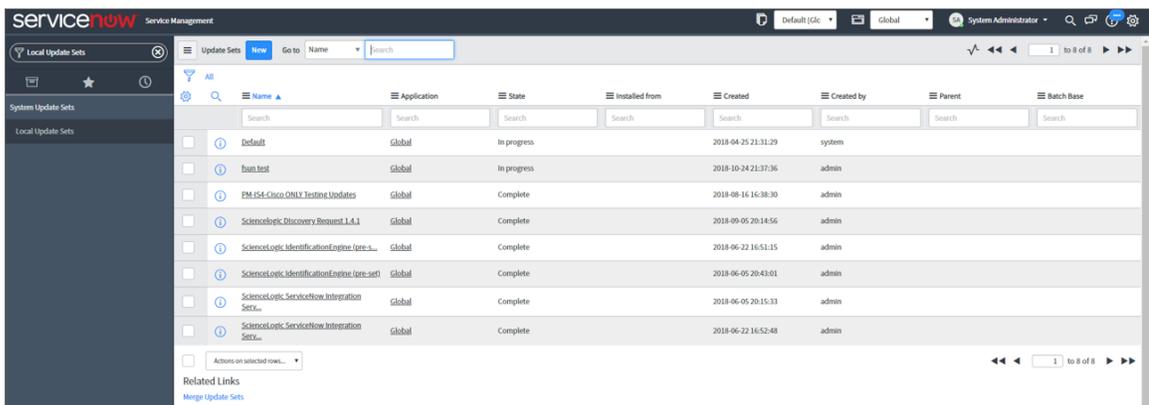
8. Click **[Update]**.
9. Repeat steps 7-8 until all relevant containment and hosting rules are in the new update set, and then go to [Exporting an Update Set](#).

Exporting an Update Set

After you have created your update set and defined the service rules, mark your update set as *Complete* and export it to an XML file.

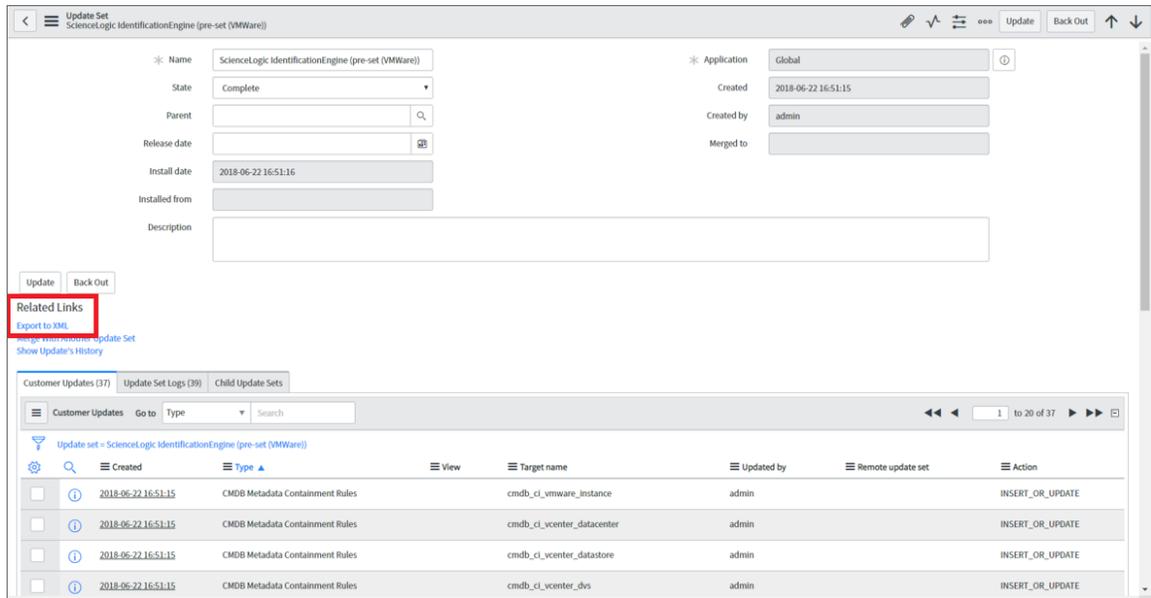
To export an update set:

1. Log in to your ServiceNow instance.
2. In the filter navigator, type “Local Update Sets” to view a list of update sets in ServiceNow:



3. Select your update set from the list.
4. Set the **State** to *Complete* and click **[Update]**.

- From the **Update Sets** page, select your completed update set from the list.
- Under the **Related Links** section, click **Export to XML**.



- Save the downloaded XML file.

Installing the ScienceLogic Domain Separation (Global) Update Set in ServiceNow

If your ServiceNow environment is **domain-separated**, where the data, processes, and administrative tasks have been organized into logical groupings called domains, you will need to install the latest version of the "ScienceLogic Domain Separation (Global)" update set in ServiceNow. This update set is *not* included in the "ScienceLogic SL1 : CMDB & Incident Automation" application (also called the Certified application).

If your ServiceNow environment does *not* use domain separation, you can skip this topic.

TIP: For more information about ServiceNow domain separation, see [Using ServiceNow Domain Separation with the Integration Service](#).

Overview of the Update Set

The "ScienceLogic Domain Separation (Global)" update set contains the following items:

- Scripted REST API
- Scripted REST Resource
- Scripted REST Query Parameter
- Scripted REST Query Parameter Association
- Script Include

This update set completely separates the ServiceNow Identification Engine REST resource that is used in the "ScienceLogic ServiceNow Integration" application and all of the required resources and duplicates it in the Global scope.

A Scripted REST Service in the Global application is a direct copy of the application endpoint with a new name: `api/10693/sciencelogic_domain_separation`. This REST Service includes only one Resource: `Device IdentificationEngine POST`. This resource works exactly like the application version, but it points to the new Script Include "SciLoDomainSepUtil". This version of the REST resource takes the same formatted JSON as the Certified application.

The Script Include "SciLoDomainSepUtil" includes all of the functionality needed to run the ServiceNow Identification Engine API.

Additional resources for the ServiceNow API:

- [CMDB Identification and Reconciliation](#)
- [identifyCI\(String jsonString\)](#)
- [createOrUpdateCI\(String source, String input\)](#)
- [Identification engine error messages](#)

NOTE: The only resource shared with this update set and the Certified application is the Device Properties page. These properties are located in the Certified application at ScienceLogic > Device > Device Properties.

Limitations of the Identification Engine

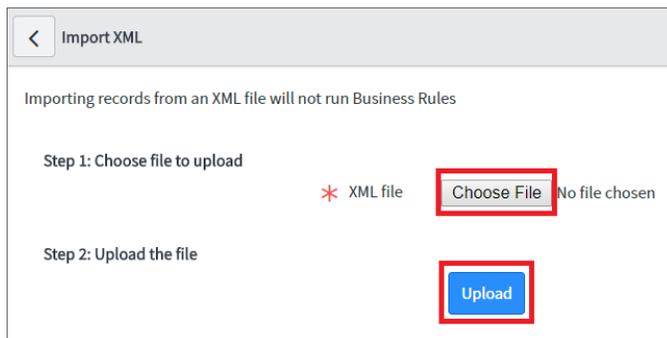
For more information about how the Identification Engine handles incoming payloads in domain-separated systems, see the following ServiceNow Knowledge Base article: [KB0695949](#).

The payload and the user domain must match, or the ServiceNow Identification Engine (IDE) will by default insert the CMDB record. Safeguards within the Integration Service Device Sync integration application were put in place for payloads that have relationships. The integration application will drop the payload if all Configuration Items do not share the same domain.

Installing the Update Set

To install the "ScienceLogic Domain Separation (Global)" update set:

1. Retrieve the latest update set from your ScienceLogic representative and download the file.
2. Log in to ServiceNow as an administrator, and then navigate to the **Retrieved Update Sets** page (System Update Sets > Retrieved Update Sets).
3. Click the **Import Update Set from XML** link under **Related Links**.
4. Click the **[Browse]** button and navigate to the update set XML file you downloaded. Select the XML file and click **[Upload]**.



5. After the file is uploaded, the **Retrieved Update Sets** page appears. Click the link for the "ScienceLogic Domain Separation (Global)" update set. The **Retrieved Update Set** page appears.
6. Click the **[Preview Update Set]** button. After the preview set runs, a status page appears.
7. Ensure that "Success" appears in the **Completion code** field.

WARNING: If "Success" does not appear in the **Completion code** field, contact ScienceLogic Support to assist with reviewing any conflicts that might exist. Do not proceed until those conflicts are resolved and "Success" appears in the **Completion code** field.

8. Click the **[Commit]** button to commit the fix script after running the preview set.
9. Before you start to sync devices, you must select the **Domain Separation** option on the **Configuration** pane in the "Sync Devices from SL1 to ServiceNow" integration application. This option ensures that the Integration Service gets re-pointed to the API endpoint after you install the "ScienceLogic Domain Separation (Global)" update set. For more information, see [Running a Device Sync](#).

Using ServiceNow Domain Separation with the Integration Service

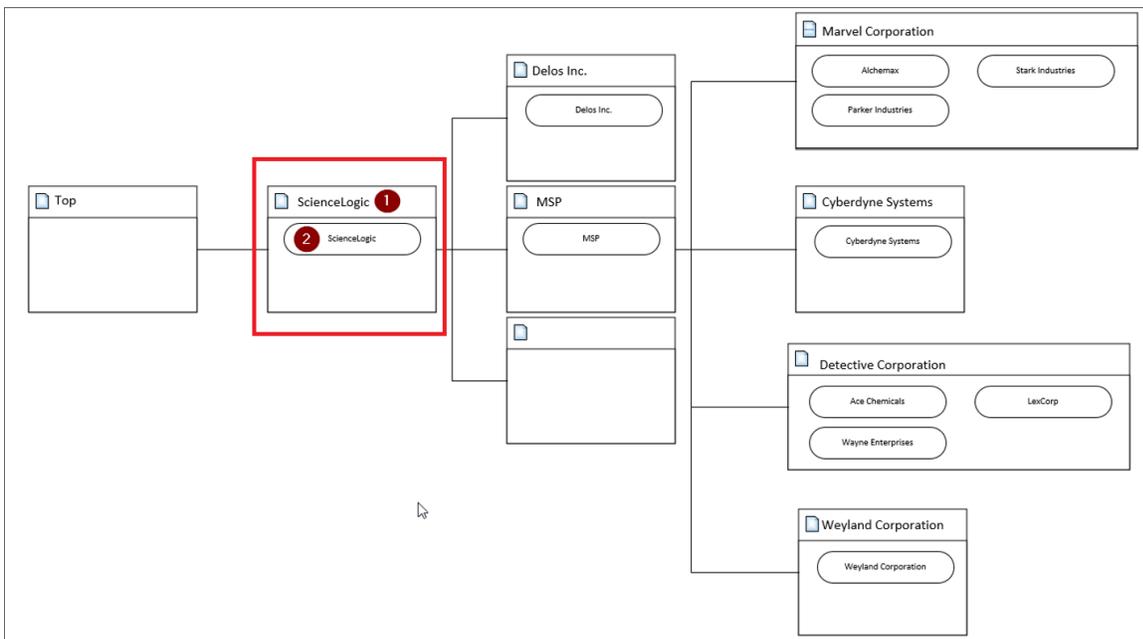
The following topics provide more information about ServiceNow domain separation and how it relates to the Integration Service. For more information, see [Domain separation](#) in the ServiceNow Documentation.

User Setup

Company and domain setup is critical for the domain separation integration to work using the Identification Engine provided by ServiceNow. This solution requires only one user and will require proper setup depending on where the user is located within the domain tree.

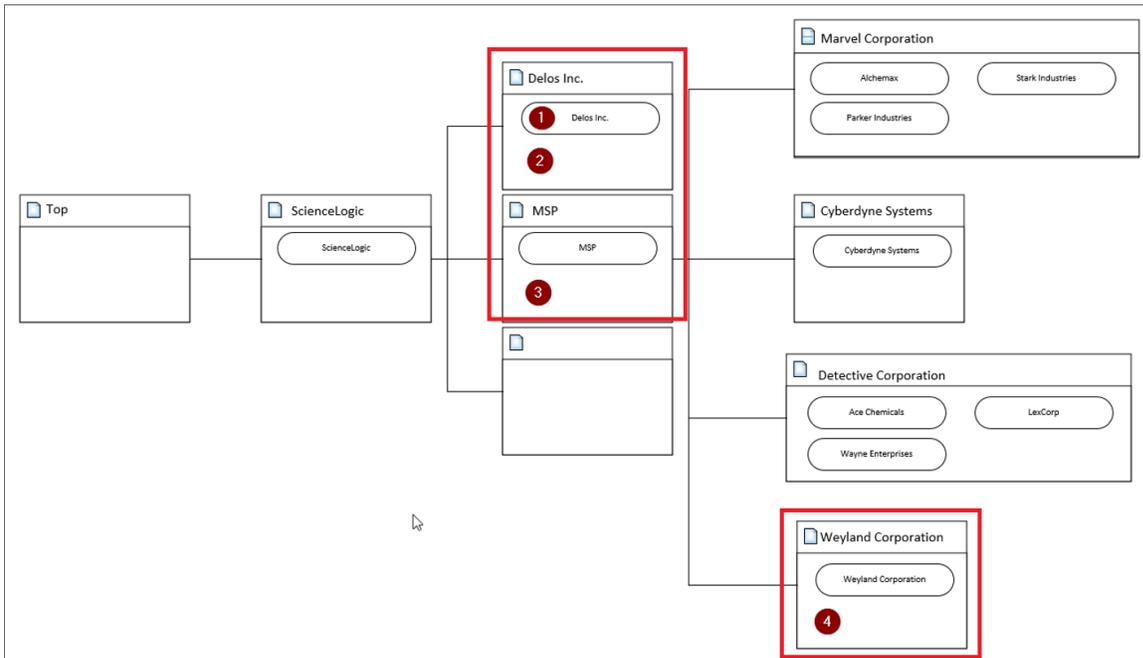
Example 1

In the following example, **ScienceLogic (1)** is both the domain and the company. The ScienceLogic user service account is associated with **ScienceLogic (2)** company, and it will have access to all child domains. You do not need to set visibility to any domain. This is the best way to set up this user, because placing it in the top domain ensures that it always has access to all children:



Example 2

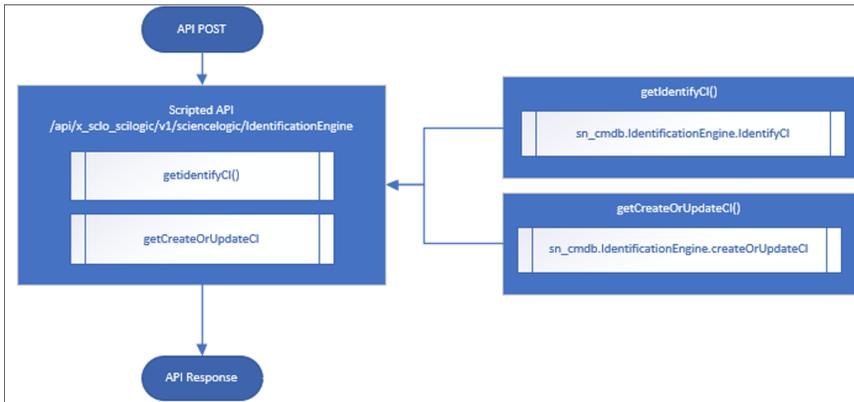
In the following example, **Delos Inc. (1)** is the company within the **Delos Inc.** domain. The Integration Service service account is associated with the **Delos Inc. (1)** company. The **Delos Inc.** domain has no children domains, and if domain visibility is not assigned, the Integration Service will not properly update the CMDB. This setup works, but it requires that proper domain visibility is set up for the service account to work correctly.



NOTE: Assigning visibility to **MSP (3)** will grant the service account access to all child domains. Assigning visibility to **Weyland Corporation (4)** will only allow access to the **Delos Inc.** domain and the **Weyland** domain; all other domains will not work.

Workflow

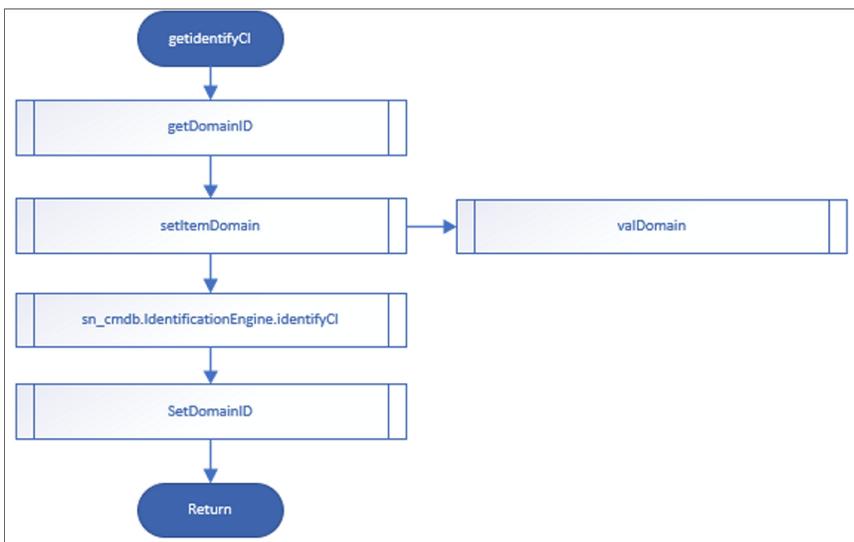
The API endpoint is based on the API query parameter "test" being true or not, which determines which Identification Engine API resource should be used. There are two primary avenues supplied with this REST resource: "createOrUpdateCI" or "identifyCI", and the only difference is that "identifyCI" does not commit the results:



The "getCreateOrUpdateCI" function uses the following workflow:

1. Retrieves the current session Domain ID (`sys_id`).
2. Sets the user Domain ID by creating an array of domain `sys_id` values and returning only the unique domains, or setting the domain if the array has only one unique domain.
3. Submits the JSON formatted string to "createOrUpdateCI()" or "identifyCI()" API.
4. Sets the user's Domain ID back to the original session ID.

The following image shows this workflow:



Syncing Organizations

If your ServiceNow configuration uses domain separation, the first sync you should run on a new Integration Service system is an **Organization Sync**. This sync uses the "Sync Organizations from SL1 to ServiceNow" integration application to sync *organizations* from SL1 with ServiceNow *companies*. Be sure to select the **Domain Separation** option on the **Configuration** pane in the "Sync Organizations from SL1 to ServiceNow" integration application. You must also select *ServiceNow* from the **Source of Truth** field.

If your ServiceNow configuration does *not* use domain separation, ScienceLogic recommends that your first sync on a new Integration Service system is an **Organization Sync** as well, but you should not select the **Domain Separation** option on the **Configuration** pane in the "Sync Organizations from SL1 to ServiceNow" integration application.

Syncing Organizations from SL1 to ServiceNow

Organization Sync uses the "Sync Organizations from SL1 to ServiceNow" integration application to pull *organizations* from SL1 and sync them with ServiceNow *companies*.

If your ServiceNow environment is domain-separated, you will need to update the following fields in ServiceNow for the companies you want to sync:

- Change the **SL1 Monitored** flag to *true* for the companies you want to sync from ServiceNow.
- Define the **SL1 Region** value to match the **region** value in the configuration object in the Integration Service user interface.
- Set the **SL1 ID** value to match the **Organization ID** in SL1.

Also, for domain-separated ServiceNow environments, you must configure and successfully run the "Sync Organizations from SL1 to ServiceNow" integration application before you can sync any additional CI items. SL1 Organizations that are linked to a ServiceNow Company will have the **crm_id** variable populated with the ServiceNow Company **sys_id** variable.

To sync SL1 organizations with ServiceNow companies:

1. In the Integration Service user interface, go to the **Integrations** page and select the "Sync Organizations from SL1 to ServiceNow" integration application. The **Integration Application** page for that application appears.
2. Click the **[Configure]** button. The **Configuration** pane appears:

Sync Organizations from SL1 to ServiceNow Compan... Cancel Save

Align configuration and save

Configuration

sl1_hostname:

snow_hostname:

sl1_user:

snow_user:

sl1_password:

snow_password:

region:

read_timeout:

Domain_Separation Update_Name

Source_of_Truth:

chunk_size: Create_Missing

3. Complete the following fields, as needed:

- **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- **region**. The region value is populated by the configuration object you selected. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration object that is aligned with this integration application, or align a different configuration object that has the correct region value.
- **Domain_Separation**. Select this option if your ServiceNow environment is *domain-separated*, where the data, processes, and administrative tasks have been organized into logical groupings called *domains*. If your ServiceNow instance is domain-separated, the user listed in the **snow_user** field must be a member of the top domain and have access to *all* of the domains you intend to integrate. Also, ServiceNow should be the "source of truth" for organizations if your environment is domain-separated.
- **Update_Name**. This option addresses the situation where the Integration Service finds a match with an organization and a company, but the names do not match. This option updates a company or organization name based on your selection in the *Source_of_Truth* field, below. For example, if you selected *ScienceLogic* as the source of truth, the Integration Service uses the company name from ScienceLogic as the updated name. This option is unselected by default.
- **Source_of_Truth**. Select whether you want to use data from ServiceNow or ScienceLogic as the "source of truth" when this integration application encounters duplicate data or data collisions.

- If you select *ServiceNow* as the source of truth, you must specify the values in the **SL1 Monitored** and **SL1 Region** fields in *ServiceNow*. Because these fields do not display by default on the **Companies** page in *ServiceNow*, navigate to the **Companies** page, click the Update Personalized List icon (), and add the **SL1 Monitored** and **SL1 Region** columns to that page. If your *ServiceNow* configuration uses domain separation, you must select *ServiceNow* as the source of truth.
 - If you select *ScienceLogic* as the Source of Truth, you do not need to do anything further.
- **Chunk_Size**. Specify the number of organizations to include in each chunk sent to *ServiceNow* when you run this integration application. The default is 500.
 - **Create_Missing**. Select this option if you want the Integration Service to create a new organization or company if that record is missing, based on your selection in the *Source_of_Truth* field. This option is unselected by default.
4. Click the **[Save]** button and close the **Configuration** pane.
 5. Click the **[Run Now]** button to run the "Sync Organizations from SL1 to *ServiceNow*" integration application.

Syncing Devices from SL1 to *ServiceNow*

The "Sync Devices from SL1 to *ServiceNow*" integration application syncs devices and virtual device relationships from SL1 to *ServiceNow*. You can also sync devices based on organization and collector group.

The Device Sync process use rules or "mappings" that you can define in the "Sync Devices from SL1 to *ServiceNow*" integration application. These mappings connect an **SL1 device class** to a ***ServiceNow* CI class**, which determines the CI class that *ServiceNow* uses when creating the CI in *ServiceNow*.

NOTE: For more information about building **service rules** (containment rules and hosting rules) for devices and CIs, see [Configuring Service Rules for Device Sync](#).

The "Sync Devices from SL1 to *ServiceNow*" integration application can also collect manufacturer and model attributes from asset records aligned with devices in SL1 and sync that information with *ServiceNow*. The Integration Service only populates the manufacturer and model attributes if the values exist in *ServiceNow* CIs; the Integration Service does not create new manufacturer values in *ServiceNow*. The "Sync Devices from SL1 to *ServiceNow*" integration application uses the **sys_id** field as a reference when syncing manufacturer and model information between SL1 and *ServiceNow*. For more information, see [Default Device Attribute Mappings](#).

Common Fields Used by Device Sync

The "Sync Devices from SL1 to *ServiceNow*" integration application uses the following fields to determine which devices to sync from SL1 to *ServiceNow*:

- **SL1 Monitored**. This field displays a Boolean (true or false) value that is impacted by whether the device is in SL1 or not. The device being found in *ServiceNow* depends on the **SL1 Monitored** field. The device being

found in SL1 depends on the class mappings defined in the "Sync Devices from SL1 to ServiceNow" integration application.

- If the CI is in ServiceNow *and* the device is in SL1, the **SL1 Monitored** flag is set to *true*.
- If the CI is in ServiceNow but the device is *not* in SL1, the **SL1 Monitored** field is set to *false*.
- **SL1 Region**. This field represents an ID for the SL1 instance or instances being synced to the ServiceNow instance. The **SL1 Region** field is determined by the user when configuring the IS applications. In a multi-SL1 environment, ScienceLogic recommends that you make the **SL1 Region** field descriptive so the ServiceNow user knows from which SL1 stack the CI originated.
 - If the **SL1 Region** field is defined as an identifier by the CI class, ServiceNow will create new CI records with the new **SL1 Region** value, and the user must manually delete the duplicate CIs in the old **SL1 Region** field.
 - If the **SL1 Region** field is *not* defined as an identifier by the CI class, ServiceNow will not treat these devices as new CIs, and the **SL1 Region** field will be automatically updated.

NOTE: Changing the **SL1 Region** value after an initial run of the "Sync Devices from SL1 to ServiceNow" application will have differing results depending on the service rules defined in ServiceNow that dictate reconciliation of the CI. If you change the **SL1 Region** value, you will need to run "Sync Devices from SL1 to ServiceNow" twice: once to align the CIs with the new region, and a second time to enable the Integration Service to re-cache the newly updated CIs in the region.

Running a Device Sync

To perform a Device Sync between SL1 and ServiceNow, run the following integration applications in the Integration Service user interface:

- **Cache ServiceNow CIs and SL1 Device Classes**. Reads all existing SL1 device classes and ServiceNow CI classes and caches them for the Device Sync. This application uses this data to populate the **mappings** drop-down values in the "Sync Devices from SL1 to ServiceNow" integration application.
- **Sync Devices from SL1 to ServiceNow**. Syncs devices and virtual device relationships from SL1 to ServiceNow. In previous versions, this integration application was named "ScienceLogic To ServiceNow Device Sync using GraphQL".

To sync SL1 devices with ServiceNow:

1. In the Integration Service user interface, select the "Cache ServiceNow CIs and SL1 Device Classes" integration application from the **[Integrations]** tab, align a configuration file with the application, and then click the **[Run]** button to run the application.

NOTE: You will need to run "Cache ServiceNow CIs and SL1 Device Classes" again if you make any relationship rule changes in ServiceNow.

- After the "Cache ServiceNow CIs and SL1 Device Classes" application completes, select the "Sync Devices from SL1 to ServiceNow" integration application from the **[Integrations]** tab and click the **[Configure]** button. The **Configuration** pane appears:

Sync Devices from SL1 to ServiceNow

Modify configuration and save.

Configuration
ven01770

sl1_hostname 10.2.11.154	sl1_db_host \${config.sl1_host}	snow_hostname ven01770.service-now.com
sl1_user em7admin	snow_user is4user1	sl1_password ●●●●●●●●●●●●●●●●
snow_password ●●●●●●●●●●●●●●●●	sl1_db_user root	sl1_db_password ●●●●●●●●●●●●●●●●
region del_test	Include_CUGs	Include_Orgs

Enter comma-separated numbers. Enter comma-separated numbers.

mappings

cmdb_ci_esx_resource_pool	maps to: Search options	VMware Resource Pool
cmdb_ci_esx_server	maps to: Search options	VMware Host Server
cmdb_ci_vcenter	maps to: Search options	VMware vCenter

Save

- Complete the following fields, as needed:

- Configuration.** Select the relevant configuration object with the relevant SL1 and ServiceNow credentials to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- region.** The region value is a unique identifier for the SL1 instance you are syncing with. If you are syncing multiple SL1 stacks to a single ServiceNow instance, each SL1 stack should have its own region value. The region value is populated by the configuration object you selected, and it must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration object that is aligned with this integration application, or align a different configuration object that has the correct region value.
- Include_CUGs.** If you want to include SL1 collector groups (CUG) in the device sync, add the CUG IDs from SL1 in this field, separated by commas.

- **Include_Orgs.** If you want to include SL1 Organizations in the device sync, add the Organization IDs from SL1 in this field, separated by commas.
 - **read_timeout.** Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
 - **Domain_Separation.** Select this option if your ServiceNow environment is *domain-separated*, where the data, processes, and administrative tasks have been organized into logical groupings called *domains*. If your ServiceNow instance is domain-separated, the user listed in the **snow_user** field must be a member of the top domain and have access to *all* of the domains you intend to integrate. Also, ServiceNow should be the "source of truth" for organizations if your environment is domain-separated.
 - **excluded_devices.** Type a list of comma-separated devices that you want to exclude from the device sync. Optional.
 - **lookup_chunk_size.** Specify the number of devices in each chunk when pulling the device cache from Couchbase. The default is 1000 devices.
 - **drop_sys_id.** Select this option if you want to disable sending the **sys_id** for CI identification. If you set **drop_sys_id** to true, make sure that ServiceNow can correctly identify and correlate your existing CIs with the properties that are available.
 - **chunk_size.** Specify the number of devices to include in each chunk sent to ServiceNow when you run this integration application. The default chunk size is 500 devices.
 - **sl1_url_override.** Update this field if you want to use an URL that is different from the standard SL1 URL that gets sent to the ServiceNow CI record. Optional.
 - **selected_devices.** If you want to sync a sub-set of all discovered devices, type a comma-separated list of the Device IDs for only the devices that you want to sync. If this field is blank, then the Integration Service runs a complete device sync.
 - **Simulation_Mode.** Select this option if you want to perform a simulated run of this integration application to show you the potential results of that run.
4. Scroll to the section for the **mappings** parameter and click **[Add Mapping]** to create a mapping between the SL1 device classes and the ServiceNow CI classes that you want to include in the device sync. ServiceNow CI classes display on the left, and SL1 device classes display on the right. You can map a single ServiceNow CI class with multiple SL1 devices classes.

TIP: Use the **[Tab]** button to move down through the list of options in a **Mapping** dropdown list, press **[Shift]+[Tab]** to move up, and press **[Enter]** to select a highlighted option. You can also add the mappings using [Postman](#).

NOTE: The "Sync Devices from SL1 to ServiceNow" integration application will *only* sync a device from SL1 if the device class for that device is mapped to a ServiceNow CI class in this section. The **mappings** section on the **Configuration** pane includes some best practices for mapping the two classes, but this section does not cover all technologies. Syncing additional technologies from SL1 to ServiceNow may require additional some research to understand the class structure.

5. Scroll to the section for the **additional_attributes** parameter and create a mapping for any other attributes you want to map between SL1 (the first column) and ServiceNow (the second column). For more information, see [Syncing Custom Device Attributes](#).
6. Click the **[Save]** button and close the **Configuration** pane.
7. Before you can run the Device Sync, you need to run the "Generate Required CI Relations for ServiceNow" integration application to determine if you are missing any class mappings or service rules that might be required in ServiceNow. Select the "Generate Required CI Relations for ServiceNow" integration application from the **[Integrations]** tab, click the **[Configure]** button, align a configuration object, and click **[Run Now]**.

NOTE: The Integration Service uses the mappings you configured in step 4 on the "Sync Devices from SL1 to ServiceNow" integration application, so you do not need to configure any mappings on the **Configuration** pane for the "Generate Required CI Relations for ServiceNow" integration application. Any mappings you add to the "Generate Required CI Relations for ServiceNow" Configuration pane will *overwrite* mappings in the "Sync Devices from SL1 to ServiceNow" application.

8. When the "Generate Required CI Relations for ServiceNow" application completes, review the log information in the **Step Log**. For more information, see [Log Messages for the "Generate Required CI Relations for ServiceNow" Application Integration](#).
9. Address any missing class mappings or service rules as needed. For more information on service rules, see [Creating a ServiceNow Update Set](#).
10. After all the mappings and other configurations are complete, run the "Sync Devices from SL1 to ServiceNow" integration application. If this is the first time you run this integration application, run it a second time to build the internal cache.

Adding Device Mappings with Postman

You can dynamically set the device mappings on a per-run basis using the API. You can also [persistently save device mappings with the API](#).

To add device mappings using Postman:

1. In Postman, POST the following JSON file to trigger the required integration applications in the Integration Service user interface to model SL1 devices to ServiceNow:

NOTE: This example only maps VMware device classes to ServiceNow, SL1 devices and a few Cisco Devices. If the customer environment has other device classes, then you must manually create the mappings.

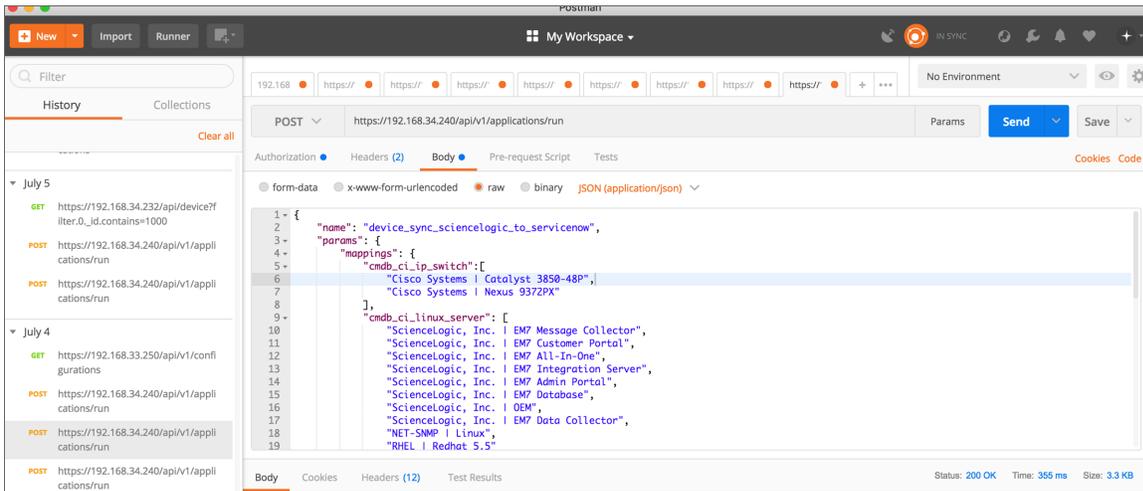
```
{
  "name": "device_sync_sciencelogic_to_servicenow",
  "params": {
    "mappings": {
      "cmdb_ci_ip_switch": [
```

```

    "Cisco Systems | Catalyst 3850-48P",
    "Cisco Systems | Nexus 9372PX"
  ],
  "cmdb_ci_linux_server": [
    "ScienceLogic, Inc. | EM7 Message Collector",
    "ScienceLogic, Inc. | EM7 Customer Portal",
    "ScienceLogic, Inc. | EM7 All-In-One",
    "ScienceLogic, Inc. | EM7 Integration Server",
    "ScienceLogic, Inc. | EM7 Admin Portal",
    "ScienceLogic, Inc. | EM7 Database",
    "ScienceLogic, Inc. | OEM",
    "ScienceLogic, Inc. | EM7 Data Collector",
    "NET-SNMP | Linux",
    "RHEL | Redhat 5.5"
  ],
  "cmdb_ci_esx_resource_pool": ["VMware | Resource Pool"],
  "cmdb_ci_esx_server": [
    "VMware | ESXi 5.1 w/HR",
    "VMware | Host Server",
    "VMware | ESX(i) 4.0",
    "VMware | ESX(i) w/HR",
    "VMware | ESX(i) 4.0 w/HR",
    "VMware | ESX(i)",
    "VMware | ESX(i) 4.1 w/HR",
    "VMware | ESXi 5.1 w/HR",
    "VMware | ESXi 5.0 w/HR",
    "VMware | ESX(i) 4.1",
    "VMware | ESXi 5.1",
    "VMware | ESXi 5.0"
  ],
  "cmdb_ci_vcenter_datacenter": ["VMware | Datacenter"],
  "cmdb_ci_vcenter_datastore": ["VMware | Datastore", "VMware | Datastore
Cluster"],
  "cmdb_ci_vcenter_dv_port_group": ["VMware | Distributed Virtual Portgroup"],
  "cmdb_ci_vcenter_dvs": ["VMware | Distributed Virtual Switch"],
  "cmdb_ci_vcenter_folder": ["VMware | Folder"],
  "cmdb_ci_vcenter_network": ["VMware | Network"],
  "cmdb_ci_vmware_instance": ["VMware | Virtual Machine"],
  "cmdb_ci_vcenter": ["VMware | vCenter", "Virtual Device | Windows Services"],
  "cmdb_ci_vcenter_cluster": ["VMware | Cluster"]
},
"configuration": "template_snow_integration" #name your configuration file
}
}

```

The following image displays an example of using Postman to send the mapping data to Integration Service:



Persistently Saving Device Mappings with the API

You can persistently save device mappings using the API.

1. Use Postman or cURL to do a GET to load the device sync integration application:

```
GET Integration_Service_hostname/api/v1/applications/device_sync_sciencelogic_to_servicenow
```

where:

- *Integration_Service_hostname* is the IP address or URL for your Integration Service system.

NOTE: The response should contain the entire JSON output for the integration application.

2. Copy the entire JSON code and save it to a file named: "device_sync_sciencelogic_to_servicenow".

- Open the new file and locate the object with the "name": "mappings" property in the "app_variables" list. The "value" property in this object specifies the mappings to use throughout the integrations:

```

"value": {
  "cmdb_ci_appl_sharepoint": [
    "VMware | Resource Pool"
  ],
  "cmdb_ci_esx_resource_pool": [
    "VMware | Resource Pool"
  ],
  "cmdb_ci_esx_server": [
    "VMware | ESXi 5.1 w/HR",
    "VMware | Host Server",
    "VMware | ESX(i) 4.0",
    "VMware | ESX(i) w/HR",
    "VMware | ESX(i) 4.0 w/HR",
    "VMware | ESX(i)",
    "VMware | ESX(i) 4.1 w/HR",
    "VMware | ESXi 5.1 w/HR",
    "VMware | ESXi 5.0 w/HR",
    "VMware | ESX(i) 4.1",
    "VMware | ESXi 5.1",
    "VMware | ESXi 5.0"
  ],
  "cmdb_ci_hyper_v_network": [
    "VMware | Resource Pool"
  ],
}

```

- Modify the "value" property of the object to use the mappings you want to use.
- Ensure that the mappings follow the same JSON data structure, otherwise the sync will not work:

```

{
  "cmdb_ci_class": [
    "ScienceLogic Dev Class | ScienceLogic subclass",
    "Another Silo Dev Class | Another Silo subclass"
  ]
}

```

- After you update the mappings, use the iscli to upload the updated integration application with your new settings. Type the following command at the command line:

```

iscli -uaf device_sync_sciencelogic_to_servicenow -H hostname_or_IP_address_of_integration_service_system -p password

```

where:

- hostname_or_IP_address_of_integration_service_system* is the hostname or IP address of the Integration Service system.
- password* is password you use to log in to the Integration Service system.

Default Device Attribute Mappings

The "Sync Devices from SL1 to ServiceNow" integration application can also collect manufacturer and model attributes from asset records aligned with devices in SL1 and sync that information with ServiceNow.

The Integration Service only populates the manufacturer and model attributes if the values exist in ServiceNow Cls; the Integration Service does not create new manufacturer values in ServiceNow. "Sync Devices from SL1 to ServiceNow" integration application uses the **sys_id** field as a reference when syncing manufacturer and model information between SL1 and ServiceNow.

The following table describes the default mappings between SL1 and ServiceNow device attributes:

SL1 attribute	ServiceNow attribute
assetTag	asset_tag
cat_name	category
cpu	cpu_count
cpu_make	cpu_type
dns_domain	dns_domain
dsk_size	disk_space
function	justification
hostname	fqdn
memory	ram
model	model_number
os	os
p_date	order_date
serial	serial_number
speed	cpu_speed
status	hardware_substatus
virtual	virtual
w_cost	cost
w_date_ex	warranty_expiration
make	manufacturer
a_notes	Not synced by default
array_size	Not synced by default
disk_count integer	Not synced by default
dns_name	Not synced by default
fw_ver (str)	Not synced by default
hostid	Not synced by default

SL1 attribute	ServiceNow attribute
is_snmp	Not synced by default
location	Not synced by default
owner	Not synced by default
purchase order number	Not synced by default
rfid	Not synced by default

NOTE: The values listed as *Not synced by default* above are queried from SL1, but those values do not have default attributes to map to in ServiceNow. As a result, the Integration Service does not sync those attributes by default. If you want to sync these SL1 attributes, you must specify the ServiceNow attributes to be synced by using the key/value mappings.

Syncing Custom Device Attributes

You can sync existing and custom attribute values for devices from ServiceNow to SL1. All custom attributes for each device are synced automatically.

NOTE: When an attribute value is "0" in SL1, the corresponding field in ServiceNow might display as empty.

To map and sync device attributes:

1. In the Integration Service user interface, go to the **[Integrations]** tab and select the "Sync Devices from SL1 to ServiceNow" integration application.
2. Click the **[Configure]** button. The **Configuration** pane appears.
3. From the **Configuration** drop-down list, select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
4. To create a custom device class or asset attribute, scroll down to the **mappings** section and click the **[Add Mapping]** button.
5. Click the mapping name to edit a device class or asset attribute in the **mappings** section. Press **[Enter]** after editing an item to make sure your changes are saved.

6. Scroll down to the **additional_attributes** section:

Sync Devices from SL1 to ServiceNow

serial + maps to: Search options + X
serial_number X

speed + maps to: Search options + X
cpu_speed X

status + maps to: Search options + X
hardware_substatus X

virtual + maps to: Search options + X
virtual X

w_cost + maps to: Search options + X
cost X

w_date_ex + maps to: Search options + X
warranty_expiration X

Add Mapping

7. To edit an existing attribute, click the attribute name and either select an attribute from the list or type a new name for the attribute. Press **[Enter]** after editing the attribute to make sure your changes are saved.

TIP: Use the **[Tab]** button to move down through the list of options in a dropdown list, press **[Shift] + [Tab]** to move up, and press **[Enter]** to select a highlighted option.

8. To create a custom attribute, click the **[Add Mapping]** button at the bottom of the section and type a name for the attribute in the first field, and then select one or more ServiceNow attributes to which the SL1 attribute should sync in the **maps to** field. Press **[Enter]** after editing the attribute to make sure your changes are saved.
9. Click the **[Save]** button and close the **Configuration** pane.
10. Click the **[Run Now]** button to run the integration application.

Adding New Device Attributes to ServiceNow

You can also add one or more new attributes to ServiceNow that you can then sync with SL1.

To add an attribute in ServiceNow:

1. In ServiceNow, search for "Tables" in the filter navigator and select **System Definition > Tables**.
2. From the **Tables** page, search for and select the table to which you want to add a field for a new attribute.
3. From the Table page, click the **[New]** button to add a new field on the table. A new record appears:

The screenshot shows the 'Dictionary Entry' form in ServiceNow. At the top, it says 'New record'. Below that, there's a help text: 'A dictionary entry manages how ServiceNow stores data in tables and fields (columns). For new dictionary entries, select a Table and the field Type of the new column. Also enter a column label, which becomes the field label, and the column name. If necessary, set a Max length for text String type fields, make the field Mandatory to save a record, and make the field a Display Value for reference fields so it appears on records that reference this table. [More Info](#)'. The form has several fields: 'Table' (dropdown menu with 'Configuration Item [cmdb_ci]' selected), 'Application' (dropdown menu with 'Global' selected), 'Type' (text input with a search icon), 'Column name' (text input), 'Function field' (checkbox), and 'Read only' (checkbox). Below these is a 'Default Value' section with a text input and a note: 'The Default value specifies what value the field has when first displayed.' At the bottom, there's a 'Submit' button and 'Related Links' with 'Advanced view'.

4. From the **Type** drop-down list, select the data type you want to store, such as *String*. Depending on your selection, additional required fields display:

This is a close-up of the 'Dictionary Entry' form. The 'Table' dropdown is set to 'Configuration Item [cmdb_ci]'. The 'Type' dropdown is set to 'String'. Below these, three fields are highlighted with a red box: 'Column label' with the value 'SL1 Region', 'Column name' with the value 'u_sciencelogic_region', and 'Max length' with the value '100'. There are also search and info icons next to the 'Type' dropdown.

NOTE: In the *String* example, above, **Column label** contains the text you want to display in ServiceNow, and **Column name** is the exact column name used by the Integration Service or the API.

5. Complete the required fields and any other fields as needed, and then click the **Submit** button. The field is added to ServiceNow.

Syncing CI Attributes from ServiceNow to SL1

The "Sync CI Attributes from ServiceNow to SL1" integration application imports CI attributes from ServiceNow to the relevant asset and attribute fields in SL1. The CI Sync supports assets, asset configuration, asset maintenance, location, production statuses, and custom attributes.

The "Sync CI Attributes from ServiceNow to SL1" integration application can sync the display value and **sys_id** of **Reference** fields, such as location, as well as the value and label of **Choice List** fields, such as operational_status. These values can be accessed by appending **_label** to the desired field name.

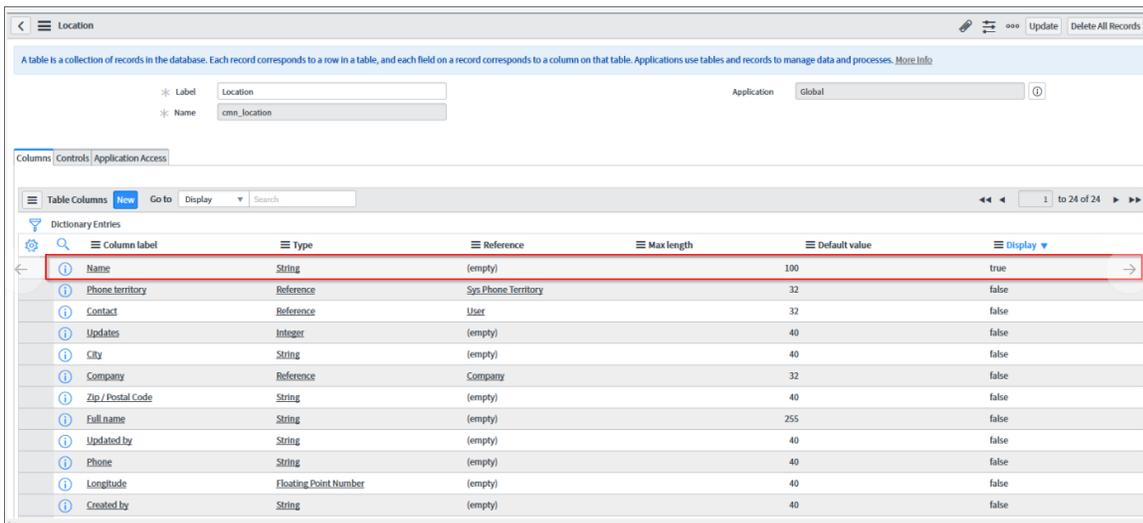
Reference Example:

```
"location": "240f6630db993300dc44f00fbf96196f"  
"location_label": "Corporate Headquarters"
```

Choice List Example:

```
"operational_status": "1",  
"operational_status_label": "Operational",
```

The following image shows the **Location** table, and the **Display** column shows the **Name** marked as **true**. Only one field on the table can be marked as **true**, and that is the field that will be returned to the Integration Service :



The screenshot shows the configuration page for the 'Location' table in ServiceNow. The 'Display' column is highlighted in red, indicating that the 'Name' field is selected for display. The table configuration is as follows:

Column label	Type	Reference	Max length	Default value	Display
Name	String	(empty)	100		true
Phone territory	Reference	Sys Phone Territory	32		false
Contact	Reference	User	32		false
Updates	Integer	(empty)	40		false
City	String	(empty)	40		false
Company	Reference	Company	32		false
Zip / Postal Code	String	(empty)	40		false
Full name	String	(empty)	255		false
Updated by	String	(empty)	40		false
Phone	String	(empty)	40		false
Longitude	Floating Point Number	(empty)	40		false
Created by	String	(empty)	40		false

NOTE: When this integration application runs, if no mappings are provided, the Integration Service queries the "Sync Devices from SL1 to ServiceNow" integration application and uses the mappings from that application.

To sync CI attributes from ServiceNow to SL1 :

1. Because this integration application uses the mappings and additional attribute options from Device Sync, go to the **[Integrations]** tab of the Integration Service user interface and run the "Sync Devices from SL1 to ServiceNow" integration application.
2. When that application completes, select the "Sync CI Attributes from ServiceNow to SL1 " integration application from the **[Integrations]** tab. The **Integration Application** page appears.
3. Click the **[Configure]** button. The **Configuration** pane appears:

Sync CI Attributes from ServiceNow to SL1

Modify configuration and save.

Configuration
cert-demo

sl1_hostname 10.2.11.154	sl1_db_host \${config.sl1_host}	snow_hostname cert023.service-now.com
sl1_user em7admin	snow_user is4user1	sl1_password ●●●●●●●●●●●●●●●●
snow_password ●●●●●●●●●●●●●●●●	sl1_db_user root	sl1_db_password ●●●●●●●●●●●●●●●●
region cert-demo	Include_CUGs Enter comma-separated numbers.	Include_Orgs Enter comma-separated numbers.

mappings

Select/type an option maps to: Search options

Add Mapping

read_timeout
20

additional_attributes

assetTag maps to: Search options

asset_tag

Save

4. From the **Configuration** drop-down list, select the configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.

NOTE: The **region** value is populated by the configuration object you selected in step 3. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you need to define the **region** variable in the configuration object that is aligned with this integration application, or align a different configuration object that has the correct region value.

5. Scroll down to the **additional_attribute** sections to edit an existing attribute, or click the **[Add Mapping]** button to create a new attribute. Press **[Enter]** after editing the attribute to make sure your changes are saved.

TIP: Use the **[Tab]** button to move down through the list of options in an **additional_attribute** dropdown list, press **[Shift]+[Tab]** to move up, and press **[Enter]** to select a highlighted option.

6. When you are done adding mappings and attributes, click the **[Save]** button and close the **Configuration** pane.
7. Click the **[Run Now]** button to run the "Sync CI Attributes from ServiceNow to SL1" integration application.

Syncing Advanced Topology Data from SL1 to ServiceNow

The "Sync Advanced Topology from SL1 to ServiceNow" integration application reads Dynamic Component Mapping relationships from SL1 and syncs those relationships with ServiceNow. If this is a new Integration Service system, you must run both the "Sync Devices from SL1 to ServiceNow" application and the "Sync Interfaces from SL1 to ServiceNow" application at least twice on new Integration Service systems to populate the cache for this integration application.

WARNING: The Integration Service only syncs topology data for devices and network interfaces that have already been synced with ServiceNow. Before setting up advanced topology sync, you must first [sync devices](#) or [sync network interfaces](#), depending on your environment.

To sync advanced topology data and relationships from SL1 to ServiceNow:

1. On the **[Integrations]** tab of the Integration Service user interface, click the **[Run Now]** button for the "Sync Devices from SL1 to ServiceNow" integration application. Run the application a second time if this is a new Integration Service system.
2. Click the **[Run Now]** button for the "Sync Interfaces from SL1 to ServiceNow" integration application. Run the application a second time if this is a new Integration Service system.
3. Select the "Sync Advanced Topology from SL1 to ServiceNow" integration application and click the **[Configure]** button on the application detail page. The **Configuration** page appears:

3. Complete the following fields, as needed:

- **Configuration.** Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- **region.** The region value is populated by the configuration object you selected. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration that is aligned with this integration application, or align a different configuration that has the correct region value.
- **Domain_Separation.** Select this option if your ServiceNow environment is *domain-separated*, where the data, processes, and administrative tasks have been organized into logical groupings called *domains*. If your ServiceNow instance is domain-separated, the user listed in the **snow_user** field must be a member of the top domain and have access to *all* of the domains you intend to integrate. Also, ServiceNow should be the "source of truth" for organizations if your environment is domain-separated. This application does not support relationships for devices across domains; all devices in a relation payload must be in the same domain.
- **chunk_size.** Specify the number of topologies to include in each chunk sent to ServiceNow when you run this integration application. The default chunk size is 500.
- **read_timeout.** Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
- **Simulation_Mode.** Select this option if you want to perform a simulated run of this integration application to show you the potential results of that run.

4. Click the **[Save]** button and close the **Configuration** pane.

5. Click the **[Run Now]** button to run the integration application.

Syncing Network Interfaces from SL1 to ServiceNow

You can map and sync network interfaces in much the same way you sync devices between SL1 and ServiceNow. You run the "Sync Interfaces from SL1 to ServiceNow" integration application, which collects interface data from ServiceNow and SL1 and runs multiple CI syncs for each interface to be synced.

WARNING: The Integration Service only syncs network interfaces that are aligned with devices that are already synced with ServiceNow. Before setting up network interface sync, you must first [sync devices between SL1 and ServiceNow](#).

To sync SL1 network interfaces with ServiceNow:

1. In the Integration Service user interface, go to the **[Integrations]** tab and select the "Sync Interfaces from SL1 to ServiceNow" integration application.
2. Click the **[Configure]** button to open the **Configuration** pane:

The screenshot shows the configuration interface for the "Sync Interfaces from SL1 to ServiceNow" integration. The configuration is for the "servicenow_syncpack_sample" configuration object. The fields are as follows:

Field Name	Value	Locked
sl1_hostname	em7.sciencelogic.com	Yes
sl1_user	em7admin	Yes
sl1_db_host	\$(config.sl1_host)	Yes
sl1_password	Yes
snow_hostname	sciencelogic.service-now.co	Yes
snow_user	is4user1	Yes
snow_password	Yes
sl1_db_user	root	Yes
sl1_db_password	Yes
region	ScienceLogic	Yes
read_timeout	20	No
Domain_Separation	<input type="checkbox"/>	No
chunk_size	500	No
adapter_sync	enabled	No
Simulation_Mode	<input checked="" type="checkbox"/>	No

3. Complete the following fields, as needed:

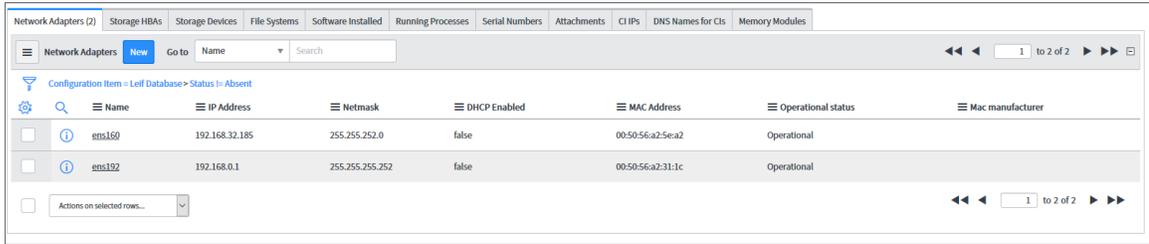
- **Configuration.** Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.

- **region**. The region value is populated by the configuration object you selected. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration object that is aligned with this integration application, or align a different configuration object that has the correct region value.
- **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
- **Domain_Separation**. Select this option if your ServiceNow environment is *domain-separated*, where the data, processes, and administrative tasks have been organized into logical groupings called *domains*. If your ServiceNow instance is domain-separated, the user listed in the **snow_user** field must be a member of the top domain and have access to *all* of the domains you intend to integrate. Also, ServiceNow should be the "source of truth" for organizations if your environment is domain-separated.
- **chunk_size**. Specify the number of devices to include in each chunk sent to ServiceNow when you run this integration application. The default chunk size is 500 devices.
- **adapter_sync**. Select one of the following settings:
 - **off**. Disables interface sync.
 - **all**. Syncs every interface, regardless of its state.
 - **enabled**. Syncs only the interfaces that have a state of "admin up". This is the default setting.
- **Simulation_Mode**. Select this option if you want to perform a simulated run of this integration application to show you the potential results of that run.

4. Click the **[Save]** button at the top of the **Configuration** pane to save your updates, and then click the **[Run Now]** button on the **Integration Application** page to run the integration application.
5. When the application completes, go to ServiceNow and type "cmdb_ci_network_adapter.list". The **Network Adapters** page appears, with a list of synced interfaces:

Name	MAC Address	IP Address	Netmask	Configuration Item	Mac manufacturer	DHCP Enabled	Status	SLI Monitored	Alias	Description	Discovery source
ems160	00:50:56:a2:3b:c8	192.168.32.187	255.255.252.0	col1		false	Installed	true		ems160	ScienceLogic
ems160	00:50:56:a2:5ea:2	192.168.32.185	255.255.252.0	Leaf Database		false	Installed	true		ems160	ScienceLogic
ems160	00:50:56:a2:45:c9	192.168.33.150	255.255.252.0	leafc2		false	Installed	true		ems160	ScienceLogic
bond0	00:50:56:a2:7b:98	192.168.32.188	255.255.252.0	192.168.32.186		false	Installed	true		bond0	ScienceLogic
ems224	00:50:56:a2:7b:98			192.168.32.186		false	Installed	true		ems224	ScienceLogic
ems192	00:50:56:a2:31:1c	192.168.0.1	255.255.255.252	Leaf Database		false	Installed	true		ems192	ScienceLogic
ems32	00:50:56:85:8e:e4	10.2.11.154	255.255.255.0	pm-abc-11-154		false	Installed	true		ems32	ScienceLogic
ems160	00:50:56:a2:08:f9	192.168.32.134	255.255.252.0	fc-dc1		false	Installed	true		ems160	ScienceLogic
ems160	00:50:56:a2:66:e4	192.168.34.242	255.255.252.0	emagpy-cmdb-34-242		false	Installed	true		ems160	ScienceLogic
ems32	00:50:56:85:57:f1	10.2.11.152	255.255.255.0	pm-abc-11-152		false	Installed	true		ems32	ScienceLogic
ems160	00:50:56:a2:7b:98			192.168.32.186		false	Installed	true		ems160	ScienceLogic
ems32	00:50:56:a2:08:c2	192.168.32.151	255.255.252.0	FC-NO-32-151		false	Installed	true		ems32	ScienceLogic
ems192	00:50:56:a2:01:88	192.168.0.2	255.255.255.252	192.168.32.186		false	Installed	true		ems192	ScienceLogic

6. Select a network interface from the list and scroll down to the **Network Adapters** tab to see more information about the interface, such as the **Operational status** value, which is synced from SL1.



	Name	IP Address	Netmask	DHCP Enabled	MAC Address	Operational status	Mac manufacturer
<input type="checkbox"/>	ens160	192.168.32.185	255.255.252.0	false	00:50:56:a2:5e:a2	Operational	
<input type="checkbox"/>	ens192	192.168.0.1	255.255.255.252	false	00:50:56:a2:31:1c	Operational	

NOTE: The **Operational status** value is different from the **SL1 Monitored** value, but the Integration Service tracks both values.

Syncing File Systems from SL1 to ServiceNow

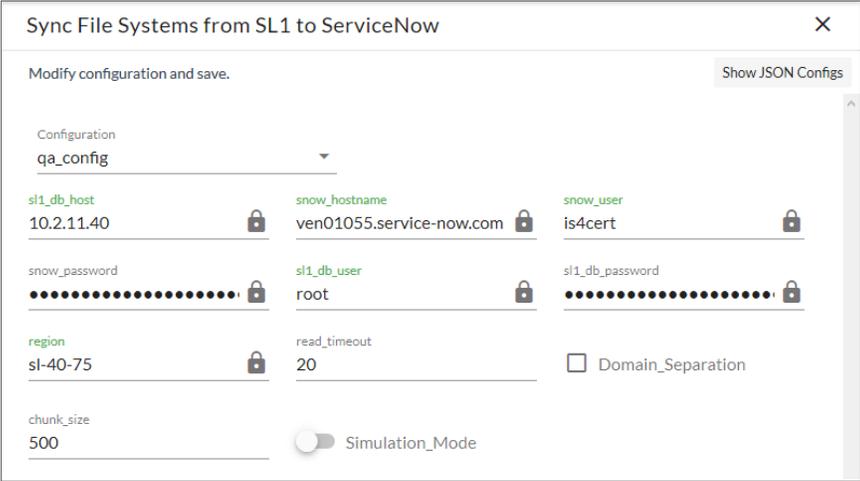
You can map and sync file systems in much the same way you sync devices between SL1 and ServiceNow. The "Sync File Systems from SL1 to ServiceNow" integration application reads file systems discovered in SL1 and then maps them to a parent CI record in ServiceNow.

WARNING: The Integration Service only syncs file systems that are aligned with devices that are already synced with ServiceNow. Before setting up file system sync, you must first [sync devices between SL1 and ServiceNow](#).

To sync SL1 file systems with ServiceNow:

1. In the Integration Service user interface, go to the **[Integrations]** tab and select the "Sync File Systems from SL1 to ServiceNow" integration application.

2. Click the **[Configure]** button to open the **Configuration** pane:



Sync File Systems from SL1 to ServiceNow

Modify configuration and save. [Show JSON Configs](#)

Configuration
qa_config

sl1_db_host 10.2.11.40	snow_hostname ven01055.service-now.com	snow_user is4cert
snow_password	sl1_db_user root	sl1_db_password
region sl-40-75	read_timeout 20	<input type="checkbox"/> Domain_Separation
chunk_size 500	Simulation_Mode <input type="checkbox"/>	

3. Complete the following fields, as needed:

- **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- **region**. The region value is populated by the configuration object you selected. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration that is aligned with this integration application, or align a different configuration that has the correct region value.
- **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
- **Domain_Separation**. Select this option if your ServiceNow environment is *domain-separated*, where the data, processes, and administrative tasks have been organized into logical groupings called *domains*. If your ServiceNow instance is domain-separated, the user listed in the **snow_user** field must be a member of the top domain and have access to *all* of the domains you intend to integrate. Also, ServiceNow should be the "source of truth" for organizations if your environment is domain-separated.
- **chunk_size**. Specify the number of file systems to include in each chunk sent to ServiceNow when you run this integration application. The default chunk size is 500.
- **Simulation_Mode**. Select this option if you want to perform a simulated run of this integration application to show you the potential results of that run.

4. Click the **[Save]** button and close the **Configuration** pane.
5. Click the **[Run Now]** button to run the integration application.

Syncing Business Services from SL1 to ServiceNow

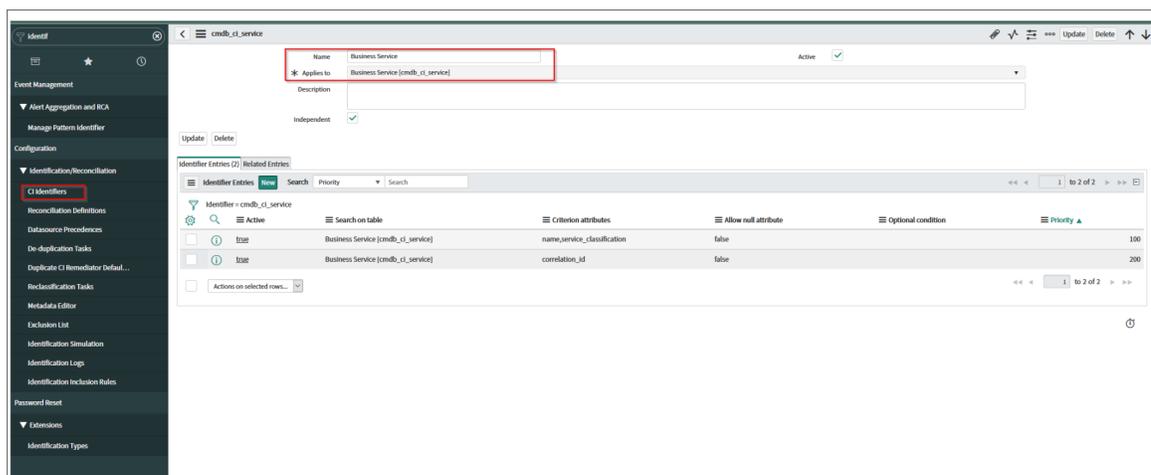
The *Sync Business Services from SL1 to ServiceNow* integration application reads Business Services, IT Services, and Device Services from SL1 and syncs them with business services in ServiceNow. This integration application creates and updates services, but it does not delete services.

NOTE: Applications and Application Components from SL1 are *not* synced by the Integration Service from SL1 to ServiceNow.

WARNING: The Integration Service only syncs business services that are aligned with devices that are already synced with ServiceNow. Before setting up business service sync, you must first *sync devices between SL1 and ServiceNow*.

To sync SL1 business services with ServiceNow:

1. In ServiceNow, create an identifier rule for syncing services by typing "CI Identifiers" in the filter navigator and clicking **[New]** on the **Identifiers** page:



2. Complete the following fields:
 - **Name.** Type a relevant name for this rule, such as "Business Service".
 - **Applies to.** Select `cmdb_ci_service`.
 - **Independent.** Select this option.
3. Right-click the gray header and click Save to save the record.
4. On the **[Identifier Entries]** tab, click **[New]** and add the relevant values from the **Criterion attributes** field for this business service, such as `name`, `service_classification` and `correlation_id`.
5. Click **[Submit]**.
6. Repeat steps 4-5 for each identifier you want to add.

10. Click the **[Save]** button and close the **Configuration** pane.
11. Click the **[Run Now]** button to run the integration application.

Syncing Installed Software between SL1 and ServiceNow

You can use the following integration applications to sync your installed software assets between and ServiceNow:

- "Sync Software Packages from SL1 to ServiceNow". Reads all software packages from SL1 and creates new CIs in ServiceNow. Run this integration before running the "Sync Installed Software" integration application.
- "Sync Installed Software from SL1 to ServiceNow". Reads all available software packages from ServiceNow and the devices aligned to that software by region and syncs them with SL1.

The integration applications do not currently support domain separation.

NOTE: The Software Asset Management (SAM) application in ServiceNow is not supported with the current level of installed software data acquired with SL1. As a result, syncing installed software data with ServiceNow Discovery and other Software Asset Management software is not currently supported.

To sync installed software between SL1 and ServiceNow:

1. Make sure that you have recently run the "Sync Devices from SL1 to ServiceNow" integration application to populate the device cache.
2. In the Integration Service user interface, go to the **[Integrations]** tab and select the "Sync Software Packages from SL1 to ServiceNow" integration application.
3. Click the **[Configure]** button to open the **Configuration** pane:

Field	Value	Locked
Configuration	ven01770	
sl1_db_host	\${config.sl1_host}	Yes
sl1_db_password	Yes
sl1_db_user	root	Yes
chunk_size	500	
snow_hostname	ven01770.service-now.com	Yes
snow_password	Yes
snow_user	is4user1	Yes
read_timeout	20	

4. Complete the following fields, as needed:
 - **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.

- **chunk_size**. Specify the number of services to include in each chunk sent to ServiceNow when you run this integration application. The default chunk size is 500.
 - **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
5. Click the **[Save]** button and close the **Configuration** pane.
 6. Click the **[Run Now]** button to run the integration application.
 7. After the "Sync Software Packages from SL1 to ServiceNow" integration application finishes running, go to the **[Integrations]** tab and select the "Sync Installed Software from SL1 to ServiceNow" integration application.
 8. Click the **[Configure]** button to open the **Configuration** pane:

Sync Installed Software from SL1 to ServiceNow

Modify configuration and save.

Configuration
ven01770

sl1_db_host \${config.sl1_host}	snow_hostname ven01770.service-now.com	snow_user is4user1
sl1_db_host \${config.sl1_db_host}	snow_hostname \${config.snow_host}	snow_user \${config.snow_user}
snow_password ●●●●●●●●●●	sl1_db_user root	sl1_db_password ●●●●●●●●●●
	sl1_db_user \${config.sl1_db_user}	
region del_test	chunk_size 500	read_timeout 20
region \${config.region}		

9. Complete the following fields, as needed:
 - **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
 - **region**. The region value is populated by the configuration object you selected. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration that is aligned with this integration application, or align a different configuration that has the correct region value.
 - **chunk_size**. Specify the number of services to include in each chunk sent to ServiceNow when you run this integration application. The default chunk size is 500.
 - **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
10. Click the **[Run Now]** button to run the integration application.

Discovery Sync

The Discovery Sync integration lets you use SL1 for discovering and syncing ServiceNow devices. With Discovery Sync, you start an SL1 discovery session from ServiceNow and then sync the newly discovered SL1 devices or virtual devices and their data with ServiceNow.

Before running a Discovery Sync session, you must complete the following steps first:

1. For domain-separated ServiceNow instances, perform a company sync by running the "Sync Organizations from SL1 to ServiceNow" integration application in the Integration Service user interface. For more information, see [Syncing Organizations from SL1 to ServiceNow](#).
2. In ServiceNow, configure a service request for Discovery Sync. For more information, see [Configuring a ServiceNow Service Request for Discovery Sync](#).
3. In the Integration Service user interface, run the integration applications listed in the [Discovery Sync Workflow](#).

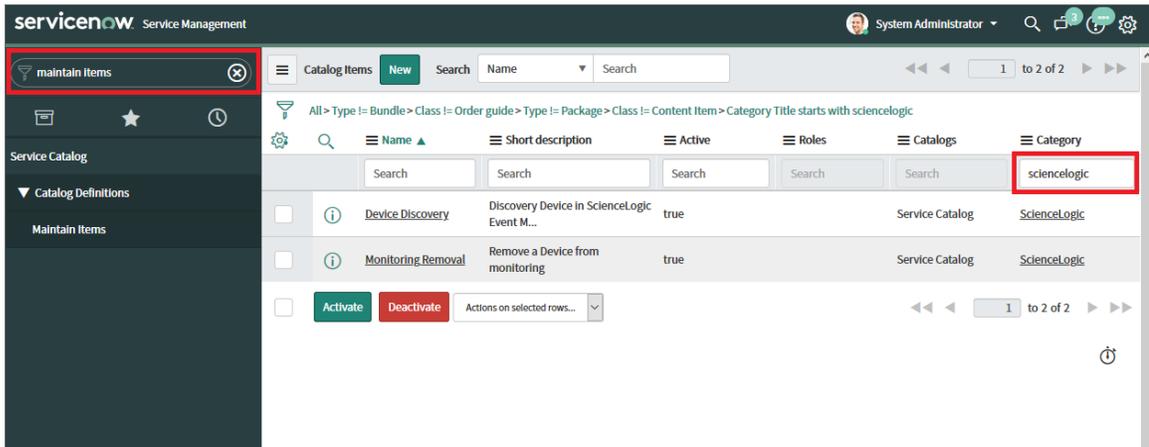
Configuring a ServiceNow Service Request for Discovery Sync

Before you can run a Discovery Sync, you need to configure the catalog and category values in the ServiceNow service request forms. You also need to activate the "Device Discovery" service request in ServiceNow.

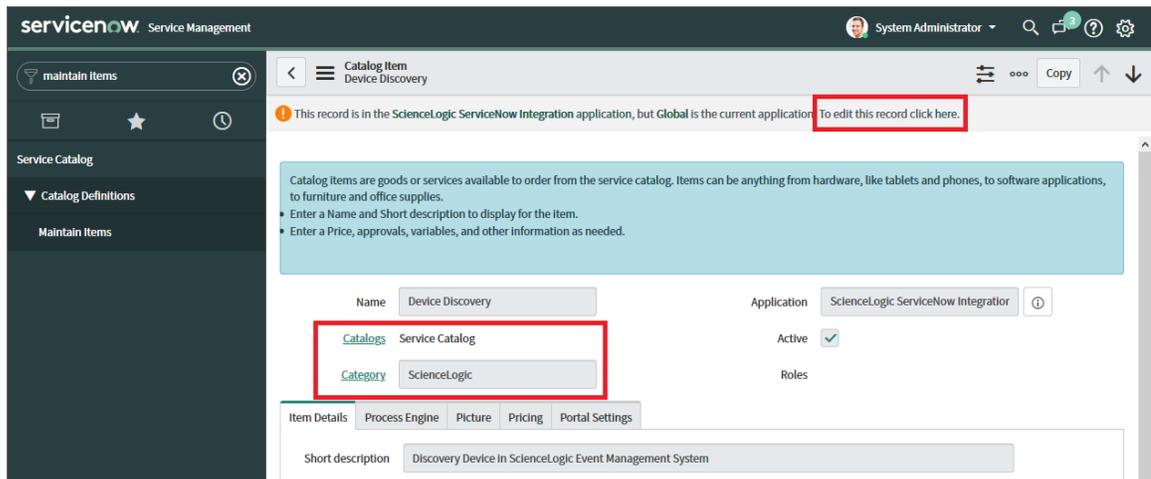
NOTE: Because some of the fields in the service request form will only populate if you have completed the previous fields in the form, you need to complete the fields in the service request form in sequential order.

To configure the ServiceNow service requests for Discovery Sync:

1. In ServiceNow, search for "Maintain Items" in the filter navigator.
2. Go to **Service Catalog > Catalog Definitions > Maintain Items** and type "ScienceLogic" in the Category field. The **Device Discovery** and **Monitoring Removal** service requests appear:



3. Open the **Device Discovery** service request and ensure that the **Catalogs** and **Category** fields are accurate. For example:



NOTE: Do not set the **Category** to a *Change Request*.

4. If you need to update these fields, click the "To edit this record click **here**" link at the top of the detail page.
5. Update the fields and click the **[Update]** button to save your changes.
6. From the **Catalog Items** page, click the check box for the **Device Discovery** service request and click **[Activate]**.

NOTE: This service request is instance-specific, which means that the service request will appear in the same location as the catalogs you specified for that request. In the example, above, the **Catalog** was set to *Service Catalog*.

7. Navigate to the relevant catalog for the service request. For example, if you selected *Service Catalog* for one or both requests, then type "Service Catalog" in the filter navigator, or select **Self-Service > Service Catalog** to view the new service requests. Type "device discovery" in the **Search catalog** field to quickly locate the request.
8. Run the integration applications listed in the [Discovery Sync Workflow](#) before creating the Device Discovery service request in ServiceNow.

Discovery Sync Workflow

To prepare SL1 and ServiceNow for a Discovery Sync, run the following integration applications in the Integration Service user interface, in the following order:

1. **Sync Discovery Requirements.** This application exports information from SL1 to populate the information in the ServiceNow request form. You must run this application before you can create the discovery sync session

in ServiceNow. This application uses one or more of the following options from the **Configuration** pane:

- **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- **region**. The region value is populated by the configuration you selected. The region value must match the value in the **SL_Region** field in ServiceNow. If you need to update this value, you will need to define the **region** variable in the configuration that is aligned with this integration application, or align a different configuration that has the correct region value.
- **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
- **Chunk_Size**. Specify the number of devices to include in each chunk sent to ServiceNow when you run this integration application. The default is 500.
- **Domain_Separation**. Select this option if your ServiceNow environment is *domain-separated*, where the data, processes, and administrative tasks have been organized into logical groupings called *domains*. If your ServiceNow instance is domain-separated, the user listed in the **snow_user** field must be a member of the top domain and have access to *all* of the domains you intend to integrate. Also, ServiceNow should be the "source of truth" for organizations if your environment is domain-separated.
- **Update_Name**. This option addresses the situation where the Integration Service finds a match with a device or CI, but the names do not match. This option updates a device or CI name based on your selection in the *Source_of_Truth* field, below. For example, if you selected *ScienceLogic* as the source of truth, the Integration Service uses the device name from *ScienceLogic* as the updated name.
- **Source_of_Truth**. Select whether you want to use data from ServiceNow or *ScienceLogic* as the "source of truth" when this integration application encounters duplicate data or data collisions.
- **Create_Missing**. Select this option if you want the Integration Service to create a new device or CI if that record is missing, based on your selection in the *Source_of_Truth* field.
- **Sync_Empty_Groups**. Select this option if you want to sync device groups that have no devices, or device groups that have devices but no matching CIs.

2. **Sync Service Requests from ServiceNow to SL1**. This application sends the request forms to SL1. This application was called "Sync Discovery Session Requests from ServiceNow to SL1" in previous versions of the SyncPack. This application uses one or more of the following options from the **Configuration** pane:

- **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- **Open_State**. The State ID from ServiceNow that specifies which Requested Items (RITMs) to pull and process. The default is 1.
- **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
- **Closed_Success_State**. The State ID for a successfully created virtual device. The State ID for a successful run changes from 1 to 2 and then ends with 4. The default is 3.

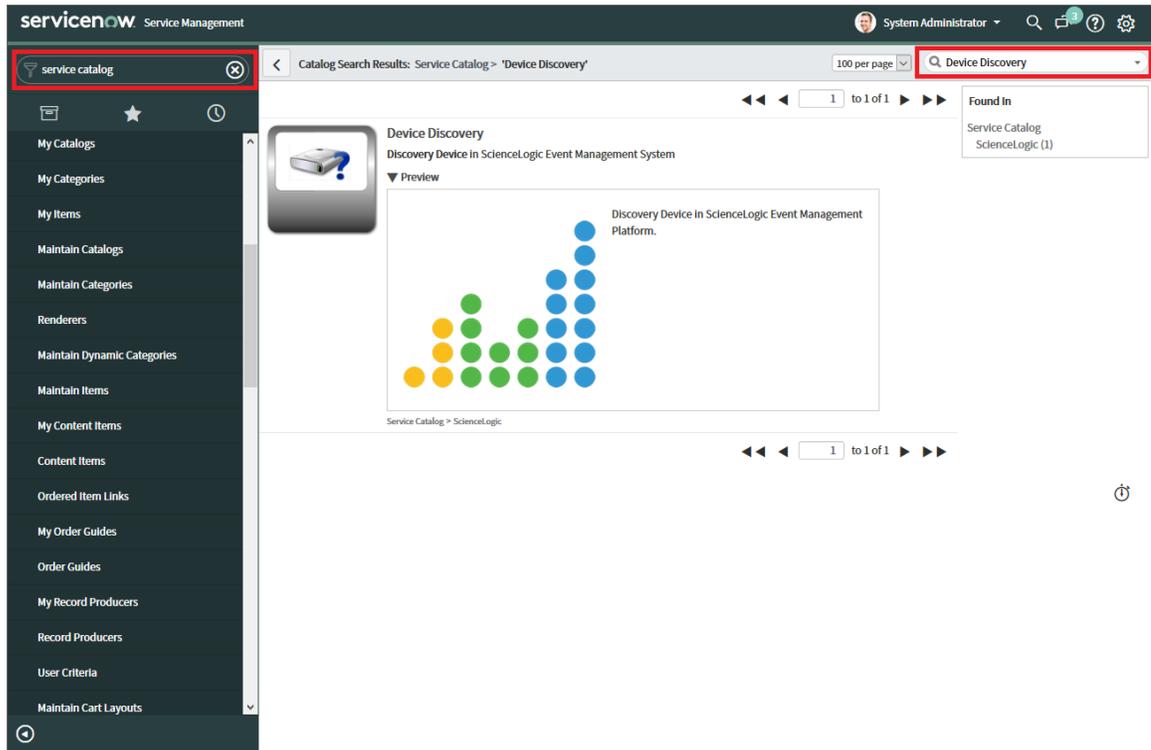
- **Closed_Failed_State** The State ID for failed discoveries or failed virtual device creation, usually caused by invalid payloads. The State ID for a failed run changes from 1 to 2 and then ends with 4. The default is 4.
 - **In_Progress_State**. The State ID for RITMs for a running discovery. The default is 2.
 - **recursively_disable_children**. Leave this field blank.
 - **target_vcug**. Leave this field blank.
3. **Sync Discovery Session Status from SL1 to ServiceNow**. This application populates the discovery session logs back to ServiceNow. This application uses the following options from the **Configuration** pane:
 - **Configuration**. Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
 - **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
 - **Closed_Success_State**. The State ID for a successfully created discovery. The State ID for a successful run changes from 1 to 2 and then ends with 4. The default is 3.
 4. **Sync Devices from SL1 to ServiceNow**. Running this application ensures that the devices discovered by SL1 get synced to ServiceNow.
 5. When the integration applications finish running, the Integration Service sends the status of those applications to ServiceNow, and you can [run a Discovery Sync in ServiceNow](#).

Running a Discovery Sync in ServiceNow

The Discovery Sync process starts an SL1 discovery session from ServiceNow and syncs the newly discovered SL1 devices and their data with ServiceNow. You can choose to discover standard devices or virtual devices.

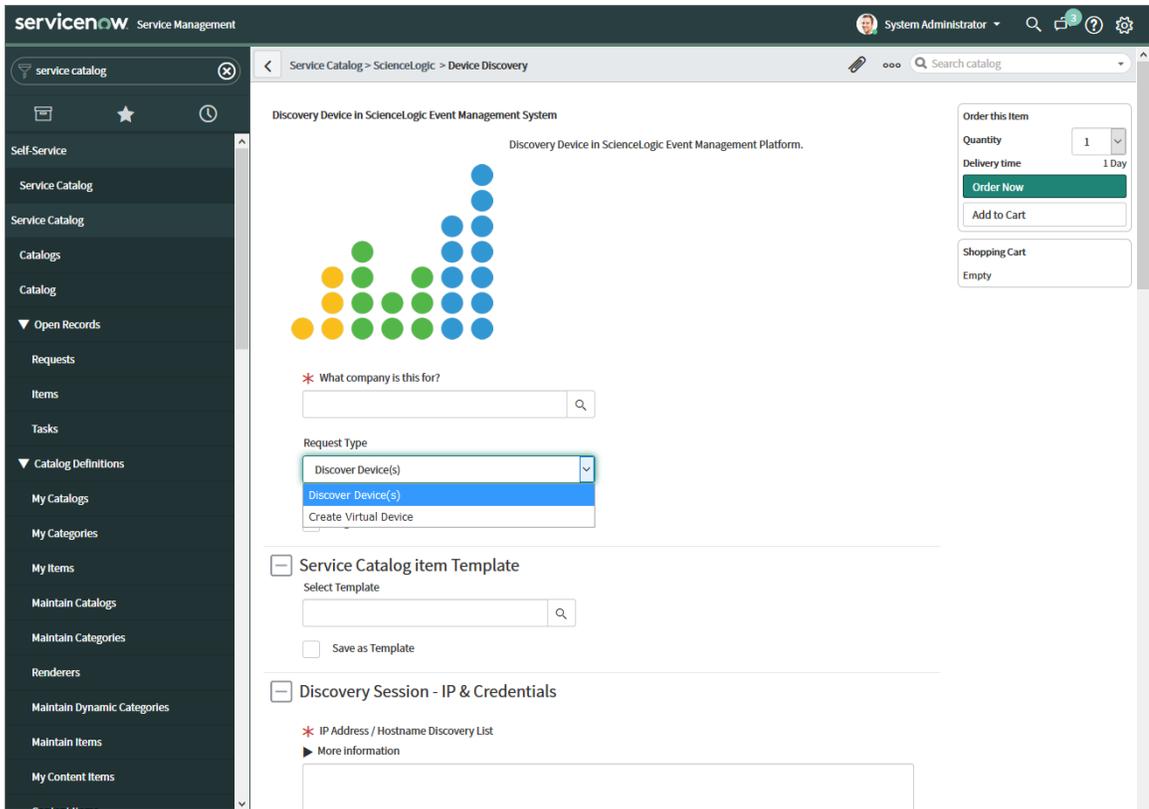
To run a Discovery Sync from the Service Catalog page:

1. In ServiceNow, search for "service catalog" in the filter navigator.
2. Navigate to the **Service Catalog** page (Self-Service > Service Catalog), type "device discovery" in the **Search catalog** field at the top right, and press **[Enter]**. The **Device Discovery** catalog entry appears:



NOTE: Previous versions of the "ScienceLogic SL1 : CMDB & Incident Automation Application" (also called the Certified or Scoped Application) created two separate service requests: **Create Virtual Device** and **Device Discovery**. Both features have been combined into the **Device Discovery** service request.

3. Click **Device Discovery**. The **Device Discovery** service request appears:



4. In the **What company is this for?** field, specify the company. The **Region** field updates automatically based on the company you select.
5. In the **Request Type** field, select *Discover Device(s)* or *Create Virtual Device*, depending on the type of device you want to discover.
 - If you selected *Discover Device(s)*, go to step 6.
 - If you selected *Create Virtual Device*, go to step 7.
6. If you selected *Discover Device(s)* in the **Request Type** field, complete the following fields:
 - **Log All.** Select this option if you want the discovery session to use verbose logging. When you select this option, SL1 logs details about each IP address or hostname specified in the **IP Address/Hostname Discovery List** field, even if the results are "No device found at this address."
 - **Select Template.** To use a template that contains your device discovery information, select the template from the dropdown.

TIP: You can save the current device discovery as a template by checking **Save as Template**. A template saves all of the discovery settings except for the IP addresses. You can access existing templates on the **Catalog Template** page in ServiceNow (ScienceLogic > Automations > Catalog Templates).

- **IP Address/Hostname Discovery List.** Provide a list of IP addresses, hostnames, or fully-qualified domain names for SL1 to scan during discovery:

- One or more *single IPv4 addresses* separated by commas and a new line. Each IP address must be in standard IP notation and cannot exceed 15 characters. For example, "10.20.30.1, 10.20.30.2, 10.20."
 - One or more *ranges of IPv4 addresses* with "-" (dash) characters between the beginning of the range and the end of the range. Separate each range with a comma. For example, "10.20.30.1 – 10.20.30.254".
 - One or more IP address ranges in *IPv4 CIDR notation*. Separate each item in the list with a comma. For example, "192.168.168.0/24".
 - One or more hostnames (fully-qualified domain names). Separate each item in the list with a comma.
- **Credentials**. Select one or more SNMP credentials to allow SL1 to access a device's SNMP data.
- **Discover Non-SNMP**. Specifies whether or not SL1 should discover devices that don't respond to SNMP requests.
- **Model Devices**. Determines whether or not the devices that are discovered with this discovery session can be managed through SL1.
- **DHCP**. Specifies whether or not the specified range of IPs and hostnames use DHCP. If you select this option, SL1 performs a DNS lookup for the device during discovery and each time SL1 retrieves information from the device.
- **Device Model Cache TTL (h)**. Amount of time, in hours, that SL1 stores information about devices that are discovered but not modeled, either because the **Model Devices** option is not enabled or because SL1 cannot determine whether a duplicate device already exists. The cached data can be used to manually model the device from the **Discovery Session** window.
- **Collection Server**. Select an existing collector to monitor the discovered devices. Required.
- **What company is this for?**. Specify the company that will use this discovery data. Click the magnifying glass icon to locate a company.
- **Add Devices to Device Groups**. Select one or more existing device groups to which you want to add the discovered devices.
- **Apply Device Template**. Select an existing device template if needed. As SL1 discovers a device in the IP discovery list, that device is configured with the selected device template.
- **Initial Scan Level**. For this discovery session only, specifies the data to be gathered during the initial discovery session.
- **Scan Throttle**. Specifies the amount of time a discovery process should pause between each specified IP address (specified in the **IP Address/Hostname Discovery List** field). Pausing discovery processes between IP addresses spreads the amount of network traffic generated by discovery over a longer period of time.
- **Scan Default Ports**. Select this option to scan the default ports: 21,22,23,25,80. If you de-select this option, you can specify a different list of ports in the **Custom Port Scan** field that appears.
- **Port Scan All IPs**. For the initial discovery session only, specifies whether SL1 should scan all IP addresses on a device for open ports.

- **Port Scan Timeout.** For the initial discovery session only, specifies the length of time, in milliseconds, after which SL1 should stop trying to scan an IP address for open ports and begin scanning the next IP address (if applicable).
- **Interface Inventory Timeout (ms).** Specifies the maximum amount of time that the discovery processes will spend polling a device for the list of interfaces. After the specified time, SL1 will stop polling the device, will not model the device, and will continue with discovery. The default value is 600,000 ms (10 minutes).
- **Maximum Allowed Interfaces.** Specifies the maximum number of interfaces per devices. If a device exceeds this number of interfaces, SL1 stops scanning the device, will not model the device, and will continue with discovery. The default value is 10,000.
- **Bypass Interface Inventory.** Select this option if you do not want SL1 to attempt to discover interfaces for each device in the discovery session.

7. If you selected *Create Virtual Device* in the **Request Type** field, complete the following fields:

- **Name.** Type a name for the virtual device.
- **Virtual Device Class.** Specify the device class of the virtual device. Click the magnifying glass icon to locate any classes aligned with your organization.
- **Collector Group.** Specify the SL1 collector group to use for the Discovery Sync. Click the magnifying glass icon to locate any collector groups aligned with your organization.

8. Click **[Order Now]**. On the **Order Status** page that appears, make a note of value in the **Request Number** field:

The screenshot shows the ServiceNow Order Status page. The 'Request Number' field is highlighted with a red box. The table below shows the order item details.

Description	Delivery Date	Stage	Price (ea.)	Quantity	Total
Discovery Device in ScienceLogic Event Management System	2019-11-16	▶ ✓✓✓	1	1	
Total					-

9. In the Integration Service user interface, go to the **[Integrations]** tab and run the "Sync Service Requests from ServiceNow to SL1" integration application.
10. When the application completes, go to **Self-Service > My Requests** in ServiceNow.
11. Click the **RITM** record link to go to the **Requested Item** page. The **State** field should update to *Closed Complete* and the request should be assigned to itself.
12. In the Integration Service user interface, go to the **[Integrations]** tab and run the "Sync Devices from SL1 to

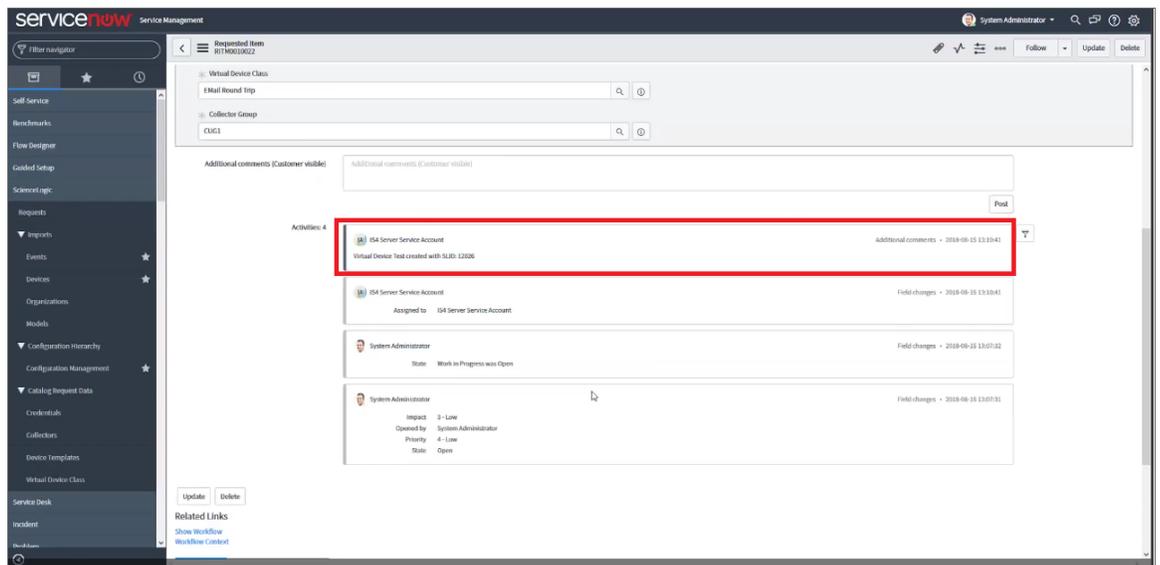
ServiceNow" integration application to make sure that the device or devices were discovered.

13. For a standard device discovery, go to ServiceNow and perform the following:

- Scroll down to the **Activities** pane to verify that you have a comment stating the discovery completed.
- In SL1, navigate to the **Discovery Control Panel** page (Registry > Manage > Discovery) and verify that SL1 created a new discovery session with that ID.

14. For a virtual device discovery, go to ServiceNow and perform the following:

- Scroll down to the **Activities** pane to verify that you have a comment stating "Virtual Device <name> Created with SLID: <new id>"



- In SL1, navigate to the **Device Manager** page (Registry > Device Manager) and verify that SL1 created a new device with that device ID.

Discovering One or More Devices from ServiceNow to SL1

If you want to quickly select one or more CIs in ServiceNow for monitoring in SL1, you can use the *Monitor Device List* option from the **Configuration Items** list view, or the *Monitor Device* option from the Configuration Item detail view.

This feature uses registered events in ServiceNow that are queued to ServiceNow Event Management to trigger actions in the Integration Service. Also, this method is just an example of one of many ways to trigger a registered event. For more information about registered events, including examples of other triggering events you can define in ServiceNow, see [Appendix D: ServiceNow Registered Events](#).

You will need to create a discovery template for a discovery process created on the **Service Catalog** page before you can discover devices using that template on the **Configuration Items** page. A template saves all of the discovery settings except for the IP addresses. You can access existing templates on the **Catalog Template** page in ServiceNow (ScienceLogic > Automations > Catalog Templates).

To discover one or more devices from ServiceNow:

1. In ServiceNow, navigate to the **Configuration Items** page.
2. From the list view, select the CI or CIs (devices) that you want to discover.

NOTE: A CI in ServiceNow must be aligned with a company in ServiceNow, or the service request will be canceled. Also, that company must be associated with a ScienceLogic Region.

3. Right-click anywhere in the window and select *Monitor Device List* from the pop-up menu. A **Select Discovery Template** dialog box appears.

TIP: You can also select a specific CI from the list view and click the *Monitor Device* option from the Configuration Item detail view. You will also need to use an existing template for this process.

4. Select a discovery template to use for the current discovery.
5. Click **[OK]** to use the template. ServiceNow generates a new service request for **Device Discovery** for each CI.
6. In the Integration Service user interface, select the "Sync Service Requests from ServiceNow to SL1" integration application and click the **[Configure]** button on the application detail page. The **Configuration** pane appears:

Field Name	Value	Locked
sl1_hostname	10.2.11.154	Yes
\$(config.sl1_host)		
snow_hostname	cert023.service-now.com	Yes
\$(config.snow_host)		
sl1_user	em7admin	Yes
\$(config.sl1_user)		
snow_user	is4user1	Yes
\$(config.snow_user)		
sl1_password	Yes
snow_password	Yes
region	cert-demo	Yes
\$(config.region)		
Open_State	1	
read_timeout	20	
Closed_Success_State	3	
Closed_Failed_State	4	
In_Progress_State	2	
target_vcug	2	
recursively_disable_children	<input type="checkbox"/>	

7. Complete the following fields, as needed:
 - **Configuration.** Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.

- **Open_State**. The State ID from ServiceNow that specifies which Requested Items (RITMs) to pull and process. The default is 1.
- **read_timeout**. Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.
- **Closed_Success_State**. The State ID for a successfully created virtual device. The State ID for a successful run changes from 1 to 2 and then ends with 4. The default is 3.
- **Closed_Failed_State**. The State ID for failed discoveries or failed virtual device creation, usually caused by invalid payloads. The State ID for a failed run changes from 1 to 2 and then ends with 4. The default is 4.
- **In_Progress_State**. The State ID for RITMs for a running discovery. The default is 2.
- **recursively_disable_children**. Leave this field blank.
- **target_vcug**. Leave this field blank.

8. Click the **[Save]** button and close the **Configuration** pane.
9. Click the **[Run Now]** button to run the integration application.
10. Go to the **[Integrations]** tab and run the "Sync Devices from SL1 to ServiceNow" integration application to make sure that the device or devices were discovered.

Decommissioning Devices

If you want to quickly select one or more CIs in ServiceNow for to remove from monitoring (or "decommission") in SL1, you can use the *Device Monitoring Removal list* option from the **Configuration Items** list view, or the *Monitoring Removal* option from the Configuration Item detail view.

You then use the "Sync Service Requests from ServiceNow to SL1" integration application to decommission the devices that you no longer want to monitor. Running this integration application takes the list of synced devices in the service request and moves them to an SL1 Virtual Collector Group (VCUG). The "Sync Service Requests from ServiceNow to SL1" integration application was formerly named "Sync Discovery Session Requests from ServiceNow to SL1".

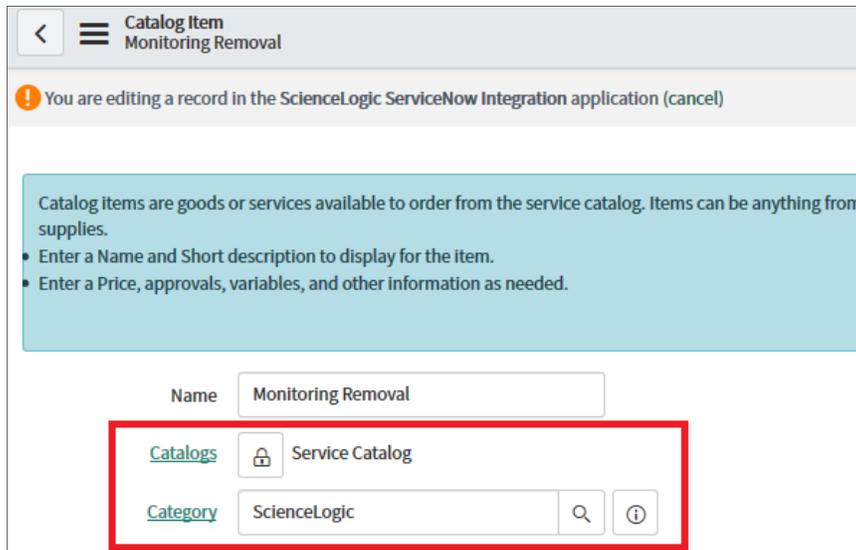
WARNING: If you move a parent device to a new VCUG, then all of its children move as well. If you move a child directly, only the child moves.

This feature uses registered events in ServiceNow that are queued to ServiceNow Event Management to trigger actions in the Integration Service. Also, this method is just an example of one of many ways to trigger a registered event. For more information about registered events, including examples of other triggering events you can define in ServiceNow, see [Appendix D: ServiceNow Registered Events](#).

Activating the ServiceNow Service Request for Monitoring Removal

To activate the ServiceNow service request for Device Decommission:

1. In ServiceNow, search for "Maintain Items" in the filter navigator.
2. Go to **Service Catalog > Catalog Definitions > Maintain Items** and type "ScienceLogic" in the Category field.
3. Open the "Monitoring Removal" service request and ensure that the **Catalogs** and **Category** fields are complete. Add the relevant information if the fields are blank. For example:



Catalog Item
Monitoring Removal

You are editing a record in the ScienceLogic ServiceNow Integration application (cancel)

Catalog items are goods or services available to order from the service catalog. Items can be anything from supplies.

- Enter a Name and Short description to display for the item.
- Enter a Price, approvals, variables, and other information as needed.

Name: Monitoring Removal

Catalogs: Service Catalog

Category: ScienceLogic

NOTE: Do not set the **Category** to a *Change Request*.

4. If you need to update these fields, click the "To edit this record click **here**" link at the top of the detail page.
5. Update the fields and click the **[Update]** button to save your changes.
6. From the **Catalog Items** page, click the check box for the **Monitoring Removal** service request and click the **[Activate]** button at the bottom of the **Catalog Items** window.
7. Navigate to the relevant catalog for the service request. For example, if you selected *Service Catalog*, then type "Service Catalog" in the filter navigator, or select **Self-Service > Service Catalog** to view the new service requests.

Removing Devices from Monitoring

To decommission Configuration Items (devices) in ServiceNow that you no longer want to monitor:

1. In ServiceNow, navigate to the **Configuration Items** window.
2. From the list view, select the CI or CIs (devices) that you want to decommission.

- **Configuration.** Select the relevant configuration object to align with this integration application. You cannot edit fields that are populated by the configuration object. Required.
- **max_age.** Specify how long (in days) that you want to keep the devices in the VCUG before deleting the devices. The default is 0 days. If this setting is 0, all devices in the VCUG will be deleted as soon as this application runs. If this setting is null, the application will fail. If all device children are in the same VCUG, the application will delete the target device and all of its children.
- **target_vcug.** Specify the ID of the SL1 Virtual Collection Group (VCUG) you created to hold the devices on the **Collector Group Settings** page (System > Settings > Collector Groups). Set this value to **-1** if you want this integration applications to use the **target_vcug** value from the "Sync Service Requests from ServiceNow to SL1" integration application.

WARNING: If you specify a value to **target_vcug** here, the "Delete Devices from SL1" application will use that value instead of the **target_vcug** value from the "Sync Service Requests from ServiceNow to SL1" application.

- **read_timeout.** Specify the maximum amount of time in seconds the integration application should wait for a response before timing out. The default is 20 seconds.

4. Click the **[Save]** button and close the **Configuration** pane.
5. Click the **[Run Now]** button to run the integration application.

Scheduling Integration Applications

Using the Integration Service user interface, you can configure integration applications to run on a schedule instead of manually running the applications. As a best practice, if you use any of these applications, ScienceLogic recommends that you schedule those applications, in the following order:

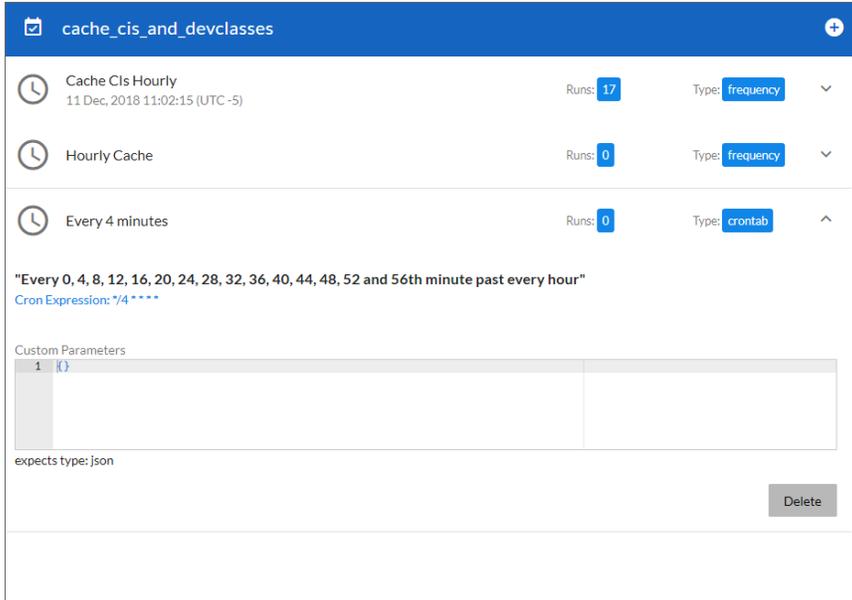
- "Cache ServiceNow CIs and SL1 Device Classes"
- "Sync Devices from SL1 to ServiceNow"
- "Sync Interfaces from SL1 to ServiceNow"

TIP: ScienceLogic recommends that you schedule these integration applications to run at least every 23 hours. You can also schedule additional applications as needed.

You can create one or more schedules for a single integration application in the Integration Service user interface. When creating each schedule, you can specify the queue and the configuration file for that integration application.

To schedule an integration application:

1. On the **Integrations** page (), click **[Schedule]** for the integration application you want to schedule. The **Schedule** window appears, displaying any existing schedules for that application:



NOTE: If you set up a schedule using a cron expression, the details of that schedule display in a more readable format in this list. For example, if you set up a cron expression of `*/4 * * * *`, the schedule on this window includes the cron expression along with an explanation of that expression: "Every 0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, and 56th minute past every hour".

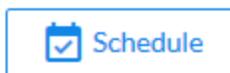
2. Select a schedule from the list to view the details for that schedule.
3. Click the + icon to create a schedule. A blank **Schedule** window appears:

The screenshot shows a 'Schedule' window with the following fields and components:

- Schedule Name:** A text input field.
- Switch to Cron Expression:** A toggle switch.
- Frequency:** A text input field with a unit of 'secs'.
- Custom Parameters:** A table with three rows of JSON data:

1	{
2	}
3	}
- expects type:** json
- Save Schedule:** A blue button at the bottom right.

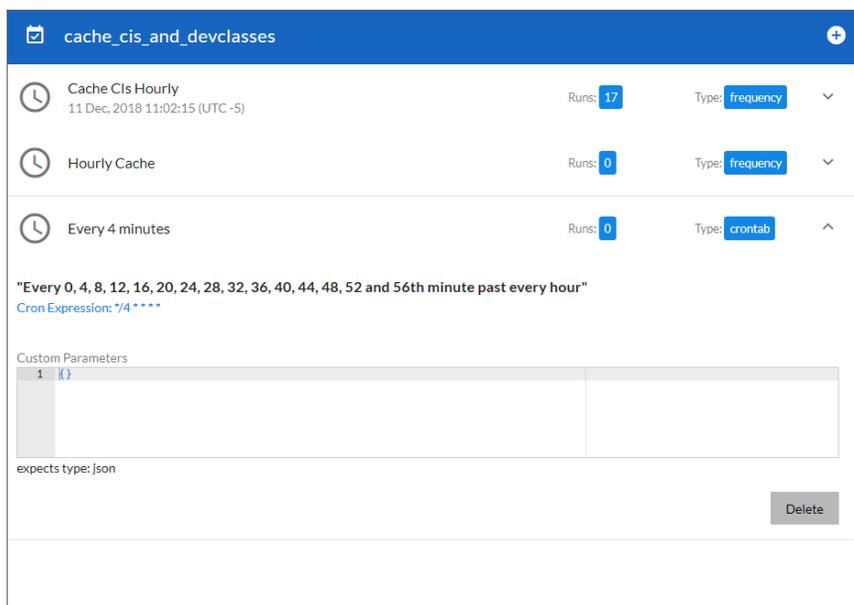
3. In the **Schedule** window, complete the following fields:
 - **Schedule Name.** Type a name for the schedule.
 - **Switch to.** Use this toggle to switch between a cron expression and setting the frequency in seconds.
 - **Cron expression.** Select this option to schedule the integration using a cron expression. If you select this option, you can create complicated schedules based on minutes, hours, the day of the month, the month, and the day of the week. As you update the cron expression, the **Schedule** window displays the results of the expression in more readable language, such as *Expression: "Every 0 and 30th minute past every hour on the 1 and 31st of every month", based on */30 */*/30 */**.
 - **Frequency in seconds.** Type the number of seconds per interval that you want to run the integration.
 - **Custom Parameters.** Type any JSON parameters you want to use for this schedule, such as information about a configuration file or mappings.
4. Click **[Save Schedule]**. The schedule is added to the list of schedules on the initial **Schedule** window. Also, on the **[Integrations]** tab, the word "Scheduled" appears in the **Scheduled** column for this integration application, and the **[Schedule]** button contains a check mark:



NOTE: After you create a schedule, it continues to run until you delete it. Also, you cannot edit an existing schedule, but you can delete it and create a similar schedule if needed.

To view or delete an existing schedule:

1. On the **Integrations** page, click **[Schedule]** for the integration application that contains a schedule you want to delete. The **Schedule** window appears.
2. Click the down arrow icon () to view the details of an existing schedule:



3. To delete the selected schedule, click **[Delete]**. The schedule is removed.

Troubleshooting CMDB Sync

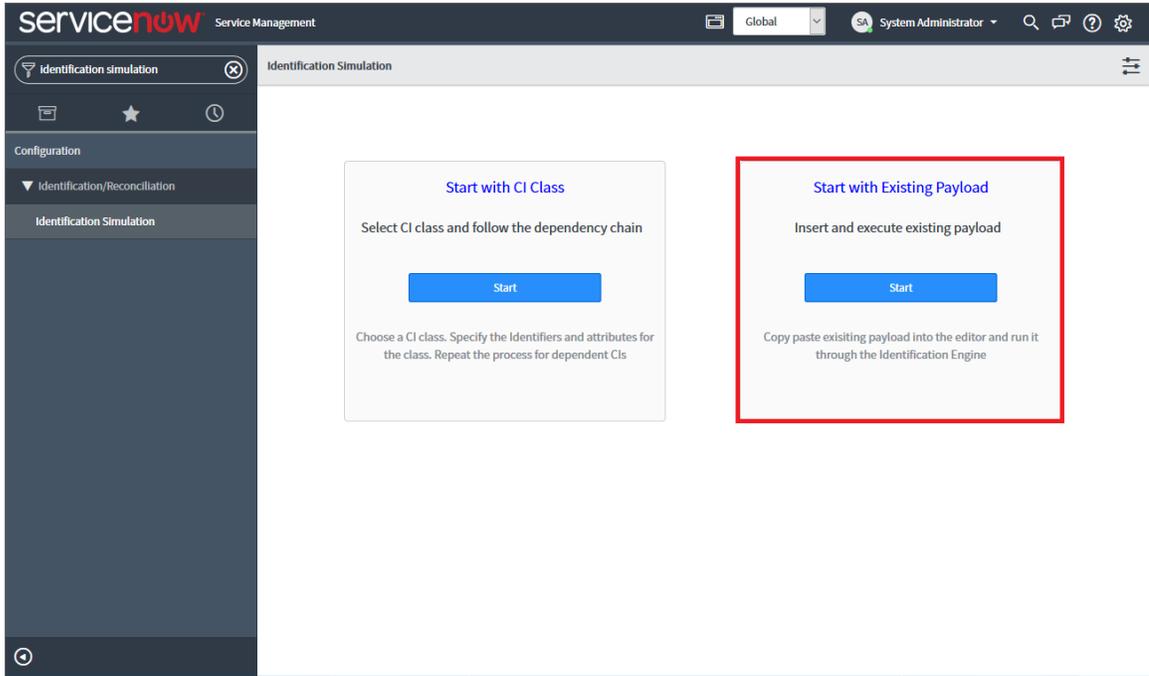
This section contains troubleshooting processes that you can use to address issues with CMDB Sync.

Issues Creating CIs in ServiceNow

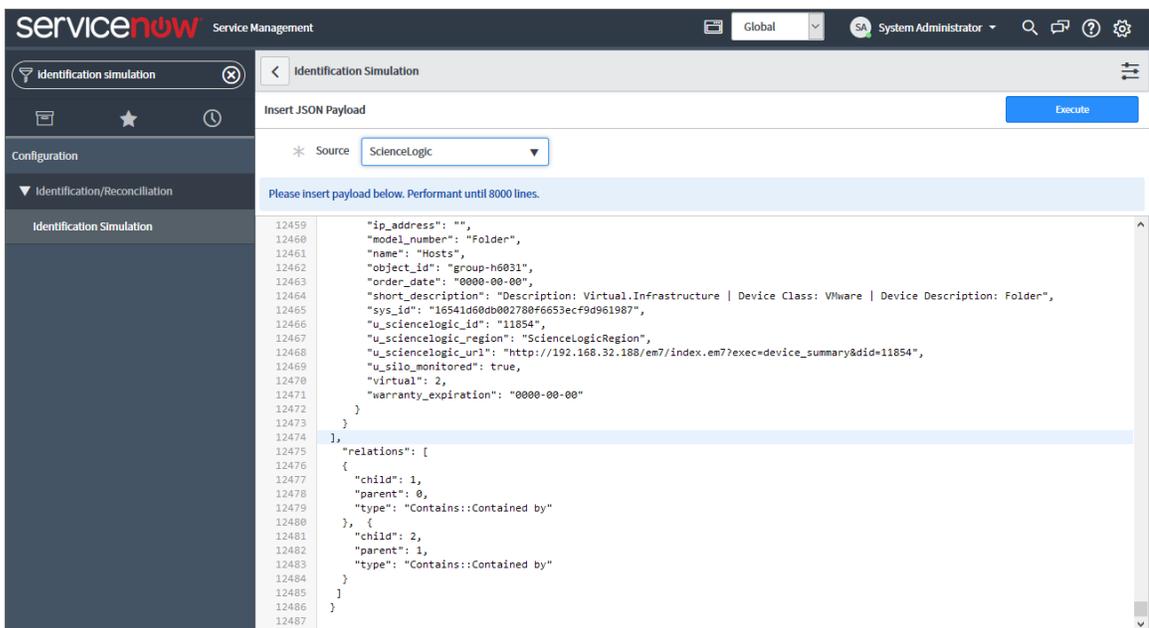
If you can successfully send data to your ServiceNow system, but you encounter issues with creating CIs in the ServiceNow CMDB, this section provides troubleshooting steps to help you test the payload and identify possible issues. These steps might be helpful if you have set up datasource precedence rules.

1. In ServiceNow, search for "import" in the filter navigator.
2. Select **ScienceLogic > Device > Imports**. The **Device Import** window appears.
3. From the list, select the Device Import log entry you want to view.

- Copy the data from the **Payload** field in the log entry and decode the data from its Base64 encoding.
- In the decoded string of data, remove the square brackets from the first and last line: ("[" , "]"")
- Copy this modified JSON payload, and then use the filter navigator to search for "Identification Simulation" or select **Configuration > Identification Simulation**.



- On the **Identification Simulation** page, click the **[Start]** button in the **Start with Existing Payload** section. The Insert JSON Payload page appears:



8. In the **Source** field, select *ScienceLogic* as the data source.
9. In the **Please insert payload below** field, paste the JSON payload you edited in step 5.
10. Click the **[Execute]** button and review the payload to identify any potential issues.

Enabling Debugging of the Configuration Item Payload

You must have administrator-level permissions in ServiceNow to access the system properties and enable debugging of the Configuration Item payload in the [ServiceNow Identification and Reconciliation module](#).

To enable debugging of the Configuration Item payload in ServiceNow:

1. On the ServiceNow system, check to see if the `glide.cmdb.logger.source.identification_engine` record exists in `sys_properties.list`.
 - If the record exists, set this value to (`*` or `debugVerbose`)
 - If the record does not exist, you will need to create the record.
2. To create the record, complete the following fields:
 - **Name.** `glide.cmdb.logger.source.identification_engine`
 - **Description.** Enable and configure the type of details the system logs when using the Identification and Reconciliation module outside the scope of identification simulation, such as when using an API, an ECC queue, or scheduled jobs (info, warn, error, debug, or `debugVerbose`).
 - **Type.** String.
 - **Value:** `*` or `debugVerbose`

NOTE: Set the system property of **Value** back to `error` when troubleshooting is complete.

3. Run the "Sync Devices from SL1 to ServiceNow" integration application. The system logs will have "identification_engine" as the source, and the log messages will contain `identification_engine` : Input.
4. Copy the payload beginning from {"items" to the end of the message. For example:

```
Message: {"items":[{"className":"","values":{"discovery_source":"ScienceLogic","mac_address":"9E:0F:04:0A:12:C7","name":"Postman Test Server 1","x_sclo_scilogic_id":"1","serial_number":"gJ3Bwkzc8r","model_id":"","ip_address":"10.10.10.102","manufacturer":"ScienceLogic, Inc.,"ram":"16000","x_sclo_scilogic_region":"Postman"},"lookup":[],"related":[]},"relations":[]}
```

5. You can run this message through the ScienceLogic endpoint by putting the {"items"} bracket within []. For example, send the following message to the endpoint

`/api/x_sclo_scilogic/v1/sciencelogic/IdentificationEngine:`

```
Message: [{"items":[{"className":"","values":{"discovery_source":"ScienceLogic","mac_address":"9E:0F:04:0A:12:C7","name":"Postman Test
```

```
Server 1", "x_sclo_scilogic_id": "1", "serial_number": "gJ3Bwkzc8r", "model_id": "", "ip_address": "10.10.10.102", "manufacturer": "ScienceLogic, Inc.", "ram": "16000", "x_sclo_scilogic_region": "Postman"}, "lookup": [], "related": []}], "relations": []}]
```

NOTE: The endpoint is different in a domain-separated environment.

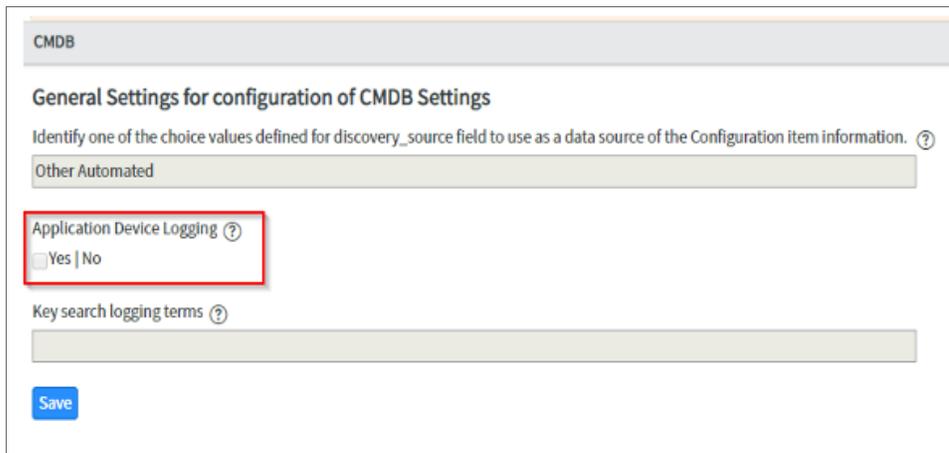
After the identification run is complete, the ServiceNow logs contain additional data about the run.

Enabling Logs in ServiceNow

You can enable device logging in ServiceNow to post errors to the logs in ServiceNow.

To enable ServiceNow device logging:

1. In ServiceNow, search for "device properties" in the filter navigator.
2. Select **ScienceLogic > Device Properties**. A **CMDB** record appears:



The screenshot shows the 'General Settings for configuration of CMDB Settings' page. The 'Application Device Logging' section is highlighted with a red box, showing a radio button for 'Yes' selected and 'No' unselected. The 'Other Automated' field is also visible.

3. Select Yes in the **Application Device Logging** section and click **[Save]**.

Log Messages for the "Generate Required CI Relations for ServiceNow" Application Integration

This section describes the different types of log messages you might see in the Step Log when you run the "Generate Required CI Relations for ServiceNow" integration application.

The following message displays if there are devices in a device tree that do not currently have a CI class mapping assigned.

```
Warning: 2751 Relations with missing mappings detected. Please re-run app with log level 10 to troubleshoot.
```

In this situation, the device tree cannot be built in ServiceNow. To address this issue, make sure that you have your entire technology tree mapped out in the **mappings** section of the "Sync Devices from SL1 to ServiceNow" integration application or in the **mappings** section of the "Generate Required CI Relations for ServiceNow" integration.

If you run the "Generate Required CI Relations for ServiceNow" integration application in Debug mode (log level 10), the application will create a log that displays the parent and child class, CI, and device ID. For example:

```
Debug: Missing Mapping for Device. Parent: {"class": "VMware | Cluster", "ci": None, "id": 76}, Child: {"class": "VMware | Host Server", ci: "cmdb_ci_esx_server", id: 363 }
```

The following message appears if the GraphQL payloads had bad data for parent and or child devices:

```
Warning: 10 bad payloads received from SL1. Re-run app in debug to troubleshoot.
```

If you run the application in Debug mode, the application will create a log that displays these payloads.

The following message appears if all relations are mapped:

```
Flow: No missing relations found!
```

The following message appears if there is a parent/child relation between ServiceNow CI classes that does not currently exist in ServiceNow and is required to sync those devices:

```
Flow: Missing Relations: [{"parent": "cmdb_ci_vcenter_folder", "child": "cmdb_ci_esx_server"}, {"parent": "cmdb_ci_vcenter", "child": "cmdb_ci_vcenter_datacenter"}]
```

Refer to the labels in the log (above) to determine which CI class is the parent type and which is the child type. To address this issue, navigate to your ServiceNow instance and create the required service rules based on the recommendations in the **Step Log**.

The following message appears if the application encounters a list of relations that are required, but were successfully found in ServiceNow:

```
Info: Found Relations: [{"parent": "cmdb_ci_vcenter_folder", "child": "cmdb_ci_esx_server"}, {"parent": "cmdb_ci_vcenter", "child": "cmdb_ci_vcenter_datacenter"}]
```

This message lets you verify that your mappings and relations are configured correctly.

Chapter

5

ServiceNow Change Management SyncPack

Overview

This chapter describes the *ServiceNow Change Management SyncPack*, which is the ScienceLogic integration with the ServiceNow Change Management Module. You can use this SyncPack to place SL1 devices into and out of maintenance mode.

This chapter covers the following topics:

<i>Downloading, Importing, and Installing the SyncPack</i>	141
<i>Integration Applications Included in the SyncPack</i>	142
<i>Syncing Device Maintenance from ServiceNow to SL1</i>	143

Downloading, Importing, and Installing the SyncPack

A SyncPack file has the **.whl** file extension type. You can download the SyncPack file from the ScienceLogic Support site.

Downloading the SyncPack

To locate and download the SyncPack:

1. Go to the ScienceLogic Support site at <https://support.sciencelogic.com/s/>.
2. Click the **Product Downloads** tab, select *PowerPacks*, and then click the "Synchronization" link. The **Synchronization PowerPack Downloads** page appears.
3. Select the SyncPack you want to install.
4. In the **Files** list, select the SyncPack **.whl** file and click **[Download]**.

NOTE: SyncPacks do not require a specific license. After you download a SyncPack, you can import it to your Integration Service using the Integration Service user interface.

Importing the SyncPack

NOTE: You must import and install the *ServiceNow Base SyncPack* before uploading and installing any of the other ServiceNow SyncPacks.

To import a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click **[Import SyncPack]**. The **Import SyncPack** page appears.
2. Click **[Browse]** and select the **.whl** file for the SyncPack you want to install.

TIP: You can also drag and drop a **.whl** file to the **SyncPacks** page.

3. Click **[Import]**. The Integration Service registers and uploads the SyncPack. The SyncPack is added to the **SyncPacks** page.

NOTE: You cannot edit the content package in a SyncPack published by ScienceLogic. You must make a copy of a ScienceLogic SyncPack and save your changes to the new SyncPack to prevent overwriting any information in the original SyncPack when upgrading.

Installing the SyncPack

To install a SyncPack in the Integration Service user interface:

1. On the **SyncPacks** page, click the **[Actions]** button () for the SyncPack you want to install and select *Activate & Install*. The **Activate & Install SyncPack** modal appears.

TIP: By default, the **SyncPacks** page only displays activated and installed SyncPacks. If you do not see the SyncPack that you want to install, click the toggle icon () on the **SyncPacks** page and select *Show All SyncPacks* to see a list of the uninstalled SyncPacks.

2. Click **[Yes]** to confirm the activation and installation. When the SyncPack is activated, the **SyncPacks** page displays a green check mark icon () for that SyncPack. If the activation or installation failed, then a red exclamation mark icon () appears.
3. For more information about the activation and installation process, click the check mark icon () or the exclamation mark icon () in the **Activated** column for that SyncPack. For a successful installation, the "Activate & Install SyncPack" integration application appears, and you can view the Step Log for the steps. For a failed installation, the **Error Logs** window appears.

Integration Applications Included in the SyncPack

This section lists the contents of the *ServiceNow Change Management SyncPack*.

Integration Applications

The following integration applications are included with the *ServiceNow Change Management SyncPack*:

- **Sync Change Management Requirements.** Sends configuration data from the Integration Service to ServiceNow to use with change management and other processes. For more information, see [Directly Enabling or Disabling Device Maintenance from ServiceNow](#).
- **Sync Maintenance Schedules from ServiceNow to SL1.** Performs maintenance of synced devices in ServiceNow and SL1. For more information, see [Scheduling Device Maintenance](#).
- **Trigger Device Maintenance Updates via MID Server.** Lets you sync maintenance windows from ServiceNow to SL1 using a ServiceNow Management, Instrumentation, and Discovery (MID) Server. For more information, see [Directly Enabling or Disabling Device Maintenance from ServiceNow](#).

Integration Applications (Internal)

To view the internal integration applications, click the Filter icon () on the **Integrations** page and select *Show Hidden Integrations*. Internal integration applications are hidden by default. The following integration applications are "internal" applications that should not be run directly, but are automatically run by applications from the previous list:

- **Cancel Maintenance**. Cancels a scheduled maintenance in SL1.
- **Create Maintenance**. Creates a scheduled maintenance in SL1.
- **Modify Maintenance**. Updates a scheduled maintenance in SL1.
- **Remove Maintenance**. Removes a scheduled maintenance in SL1.
- **Schedule Maintenance**. Creates a scheduled maintenance in SL1.

Syncing Device Maintenance from ServiceNow to SL1

You can use two different methods to put one or more devices into maintenance mode from ServiceNow to SL1:

- Use the "Sync Maintenance Schedules from ServiceNow to SL1" integration application if you want to perform *scheduled* maintenance on a set of devices in SL1. For more information, see [Scheduling Device Maintenance](#).
- Use the "Trigger Device Maintenance Updates via MID Server" integration application if you want to *immediately* enable or disable maintenance on a device. For more information, see [Directly Enabling or Disabling Device Maintenance from ServiceNow](#).

Scheduling Device Maintenance

You create a change request to perform *scheduled* maintenance through a maintenance window in ServiceNow.

WARNING: The Integration Service only syncs maintenance schedules that are aligned with devices that are already synced with ServiceNow. Before setting up maintenance schedule sync, you must first sync devices between SL1 and ServiceNow.

The CI Maintenance Sync process (non-scheduled) syncs maintenance windows from ServiceNow change requests (CHG)s to SL1 devices to place the synced devices into maintenance mode for the scheduled change window.

NOTE: The SL1 Scheduler supports maintenance windows of at least one minute or more.

To set up maintenance sync:

1. In ServiceNow, type "change" in the filter navigator and navigate to **Change > Create New**.
2. Click **[New]** to create a new change request of type "Normal". A new Change Request record appears:

The screenshot shows the ServiceNow interface for creating a new Change Request. The left sidebar is filtered to 'change' and shows the 'Create New' option. The main form is titled 'Change Request New record' and has a progress bar with stages: New, Assess, Authorize, Scheduled, Implement, Review, Closed, and Canceled. The 'New' stage is active. The form fields are as follows:

Number	CHG0030004	Type	Normal
Requested by	System Administrator	State	New
Category	Other	Conflict status	Not Run
Configuration item	gdlib-test0-cu3-34-6	Conflict last run	
Priority	4 - Low	Assignment group	Change Management
Risk	Moderate	Assigned to	
Impact	3 - Low		
Short description			
Description			

Below the form, there are tabs for Planning, Schedule, Conflicts, Notes, and Closure Information. The Planning tab is active, showing fields for Justification and Implementation plan.

3. Make a note of the change request number in the **Number** field. You will use this later to verify that the maintenance sync was created. In this example, the value is *CHG0030004*.
4. Update the following fields in the record:
 - **Configuration Item**. Select the CI you want to configure for maintenance sync.
 - **Assignment group**. Select the group for the CI.

NOTE: The aligned CI must have the **SL1 Monitored** field selected before the Integration Service can use the maintenance schedule for that CI.

- Click the **[Submit]** button. The change request is saved, and you are returned to the **Change Requests** page.
- Select the change request you just created, and in the change request record, right-click the **State** label and select *Show Choice List*. The Choices list displays a list of the configurable choices and values:

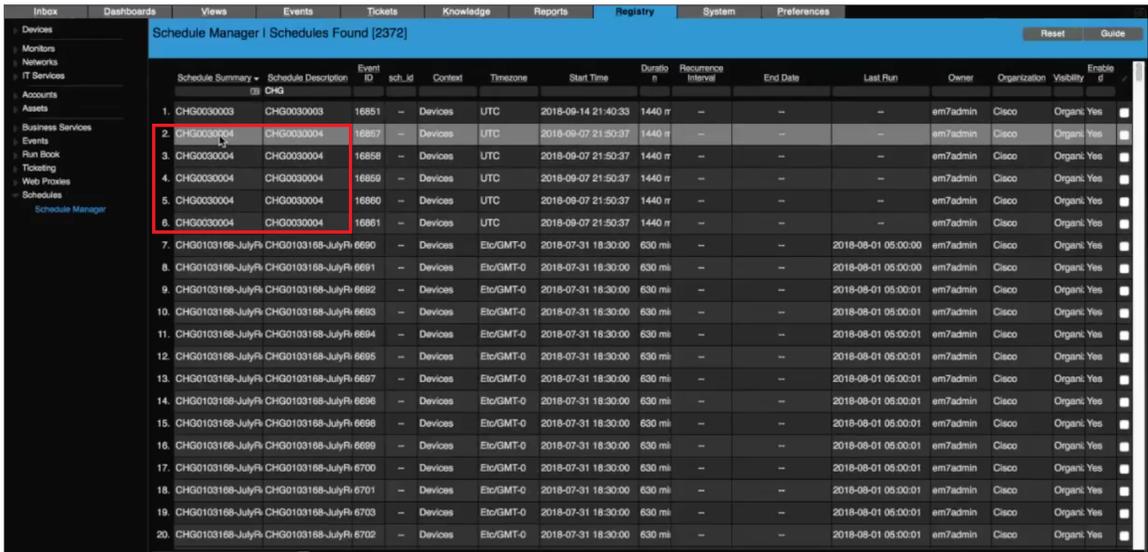
	Search	=state	Search	Search	Search	Search	Search	Search
<input type="checkbox"/>	change_request	state	en	-5	New	false	1	2015-04-24 14:29:54
<input type="checkbox"/>	change_request	state	en	-4	Assess	false	2	2015-04-24 15:07:16
<input type="checkbox"/>	change_request	state	en	-3	Authorize	false	3	2015-04-24 15:07:23
<input type="checkbox"/>	change_request	state	en	-2	Scheduled	false	4	2015-04-24 15:07:32
<input type="checkbox"/>	change_request	state	en	-1	Implement	false	5	2015-04-24 15:13:43
<input type="checkbox"/>	change_request	state	en	0	Review	false	6	2015-04-24 15:13:54
<input type="checkbox"/>	change_request	state	en	3	Closed	false	7	2015-04-24 14:31:24
<input type="checkbox"/>	change_request	state	en	4	Canceled	false	8	2015-04-24 14:32:46

NOTE: You need Administrator privileges to access this list.

- Make a note of the values in the **Value** and **Label** fields. These values map to the **New_Change_Request_State** and **Canceled_Change_Request_State** fields in the "Sync Maintenance Schedules from ServiceNow to SL1" integration application.
- Return to your new change request and scroll down in the change request to the **[Affected CIs]** tab, where you can click the **[Add]** button to add additional synced CIs to the maintenance sync:

Configuration Item	Class
gdlb-test0-cu3-34-6	Linux Server

12. Verify that the value from the **New_Change_Request_State** field matches the value in the **Value** field from ServiceNow, and the value from the **Canceled_Change_Request_State** field matches the value from the **Label** field from ServiceNow. These values must match for the maintenance sync to work.
13. Click the **[Save]** button and then click the **[Run Now]** button to run the integration application.
14. While the "Sync Maintenance from ServiceNow to SL1" integration application runs, you can monitor the status of the maintenance process by clicking the branch icon () on the **Schedule Maintenance** step. Click the triggered application's run ID in the pop-up window and then click the branch icon on the **Create SL Maintenance** or **Modify Maintenance** steps for more information.
15. After the "Sync Maintenance from ServiceNow to SL1" integration application completes, navigate to the **Schedule Manager** (Registry > Schedules > Schedule Manager) in SL1 to view the change requests.



Schedule Summary	Schedule Description	Event ID	sch. id	Context	Timezone	Start Time	Duration	Recurrence Interval	End Date	Last Run	Owner	Organization	Visibility	Enable
1. CHG0030003	CHG0030003	16851	-	Devices	UTC	2018-09-14 21:40:33	1440 rr	-	-	-	em7admin	Cisco	Organic	Yes
2. CHG0030004	CHG0030004	16857	-	Devices	UTC	2018-09-07 21:50:37	1440 rr	-	-	-	em7admin	Cisco	Organic	Yes
3. CHG0030004	CHG0030004	16858	-	Devices	UTC	2018-09-07 21:50:37	1440 rr	-	-	-	em7admin	Cisco	Organic	Yes
4. CHG0030004	CHG0030004	16859	-	Devices	UTC	2018-09-07 21:50:37	1440 rr	-	-	-	em7admin	Cisco	Organic	Yes
5. CHG0030004	CHG0030004	16860	-	Devices	UTC	2018-09-07 21:50:37	1440 rr	-	-	-	em7admin	Cisco	Organic	Yes
6. CHG0030004	CHG0030004	16861	-	Devices	UTC	2018-09-07 21:50:37	1440 rr	-	-	-	em7admin	Cisco	Organic	Yes

16. Verify that the **Schedule Summary** field contains the same value from the ServiceNow **Number** field. In this example, the value in SL1 matches the value from ServiceNow: *CHG0030004*.
17. You can also verify that the schedule was created for a device by navigating to the **Device Manager** (Registry > Devices), clicking the wrench icon for the device, and clicking the **[Schedule]** tab.
18. If you want to edit the scheduled time for the maintenance sync, open the change request in ServiceNow, click the **[Schedule]** tab, and update the **Planned start date** and **Planned end date fields** as needed. The next time the "Sync Maintenance from ServiceNow to SL1" integration application runs, the schedule is updated.
19. If you want to cancel the scheduled time for the maintenance sync, open the change request in ServiceNow, click the **Additional actions** menu button (), and select **Cancel Change**. The next time the "Sync Maintenance from ServiceNow to SL1" integration application runs, the application cancels that maintenance sync.

TIP: As a best practice, schedule the "Sync Maintenance from ServiceNow to SL1" integration application to run every hour or so, depending on your environment.

Directly Enabling or Disabling Device Maintenance from ServiceNow

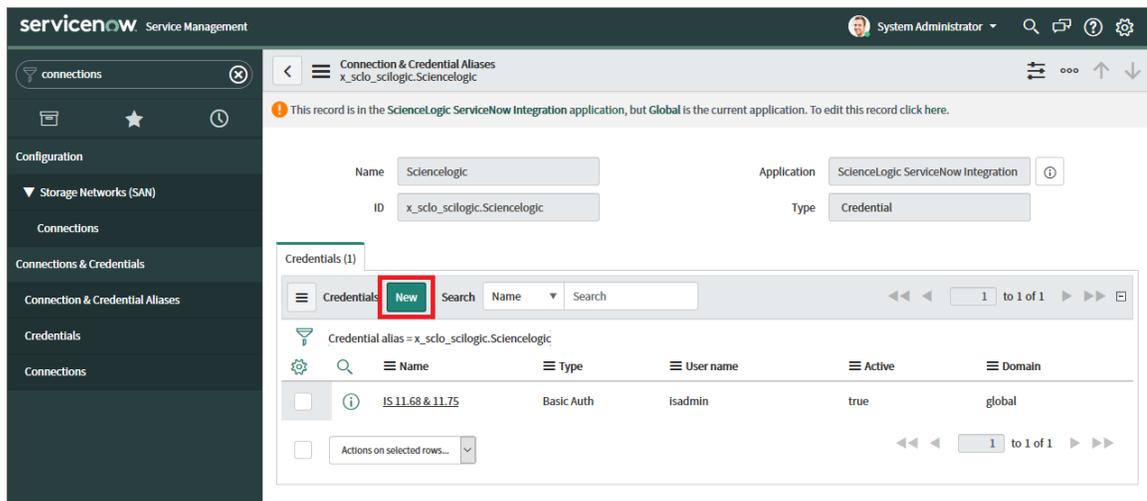
The "Trigger Device Maintenance Updates via MID Server" integration application receives a group of one or more synced devices from the ServiceNow Management, Instrumentation, and Discovery (MID) Server and checks for the *enable_maintenance* and *disable_maintenance* actions on those devices. If the application encounters devices with one of those actions, it will enable or disable the user maintenance status of those devices as needed.

You do *not* need to update any fields on the **Configuration** pane for the "Trigger Device Maintenance Updates via MID Server" integration application. You also do *not* need to run this integration application, as it is triggered by the MID Server, which is triggered first by a registered event in ServiceNow Event Management. For more information about registered events, including examples of other triggering events you can define in ServiceNow, see [Appendix D: ServiceNow Registered Events](#).

Before you can set up maintenance sync with the MID Server, you need to create a credential for the MID Server. You should have access to the "Integration Services" section of the **Discovery Dependents** page in ServiceNow.

To create a credential to connect to the Integration Service:

1. In ServiceNow, go to **Connections & Credentials > Connection & Credential Aliases**.
2. From the **Connection & Credential Aliases** list, select **ScienceLogic**. This record is provided by the Integration Service Certified Application. The **Connection & Credential Aliases** page appears:



3. Click **[New]** to create a new credential. The **Credentials** page appears.
4. From the list of credentials, select **Basic Auth Credentials**. This is currently the only type of credential that is supported. The **Basic Auth Credentials** page appears.
5. Complete the fields related to the Integration Service on the **Basic Auth Credentials** page. Make sure that the **Credential alias** field is set to *x_sclo_scilogic.ScienceLogic*.
5. Click **[Submit]**. The credential is added to the **Connection & Credential Aliases** page.
6. Select the new credential. The **Basic Auth Credentials** page for that credential appears.

7. Click **[New]** to create a Discovery IP Affinity record. A new **Credential Affinity** page appears.
8. Complete the following fields:
 - **MID server**. The name of the Mid Server you want to use.
 - **IP address**. Use the Integration Service IP address that was listed in the **IP** field on the relevant record on the **Discovery Dependents** page in ServiceNow. To quickly find the relevant record on the **Discovery Dependents** page, right-click the **Type** column and select *Group By Type*, and then expand *Type: Integration Services*. Use the IP value from the record that matches the Region for the devices you want to use.
 - **Credential ID**. This field should be completed for you.
9. Click **[Submit]**.

Appendix

A

Checklists for Deployment

Overview

This appendix describes the checklists for deploying the Integration Service and the ServiceNow SyncPacks, based on your environment and configuration.

This appendix includes the following topics:

<i>CMDB-Only ServiceNow Integration with Single SL1, no Domain Separation in ServiceNow</i>	151
<i>CMDB-Only ServiceNow Integration with Single SL1 and Domain-Separated ServiceNow</i>	154
<i>CMDB-Only ServiceNow Integration with Multiple SL1 Systems, no Domain-Separated ServiceNow</i>	158
<i>Incident-Only ServiceNow Integration with Single SL1, no Domain Separation in ServiceNow</i>	162

CMDB-Only ServiceNow Integration with Single SL1, no Domain Separation in ServiceNow

1. Deploy the Integration Service ISO:
 - IP address, Netmask, Gateway, DNS, Hostname provided
 - Root password provided (this is the root user for the OS)
 - Start Docker services after installation:

```
/opt/iservices/scripts/pull_start_iservices.sh
```
 - Validate that iservices are running:

```
docker service ls
```
2. Activate the Configuration Management For Scoped Apps (CMDB) Plugin.
3. Install the ScienceLogic Certified Application and create a ServiceNow group and user account:
 - Username
 - Password
 - Web Service Access Only
 - GMT Time Zone
 - x_sclo_scilogic.Admin role assigned
4. Install the ServiceNow SyncPacks on the Integration Service.
5. Create the Integration Service configuration object using the "ServiceNow SyncPack" configuration object as a template.

Sync Devices from SL1 to ServiceNow

1. Align the configuration object to the following integration applications:
 - Cache ServiceNow CIs and SL1 Device Classes
 - Sync Devices from SL1 to ServiceNow
2. Run "Cache ServiceNow CIs and SL1 Device Classes" to retrieve all device class information from SL1 and ServiceNow. This will populate the device class mapping in the following step. This integration should be run at least every 23 hours.
3. Configure class and attribute mappings in "Sync Devices from SL1 to ServiceNow".
4. Run "Generate Required CI Relations for ServiceNow" to see if you are missing any service rules or class mappings and create any required maps, containment rules, and hosting rules.
5. Run "Sync Devices from SL1 to ServiceNow" either manually or on a schedule. This integration should be run at least every 23 hours. See documentation for more information.

Discover Devices from ServiceNow in SL1

1. Align the configuration object to the following integration applications:
 - Sync Discovery Requirements
 - Sync Discovery Session Requests from ServiceNow to SL1
 - Sync Discovery Session Status from SL1 to ServiceNow
2. Set additional configuration variables for each of the integrations applications above in the respective **Configuration** pane.
3. Run "Sync Discovery Requirements" to sync all discovery-dependent information from SL1 to ServiceNow.
4. Make sure that the Discovery request RITM is successfully created and approved in ServiceNow using the provided Service Catalogs.
5. Run "Sync Discovery Session Requests from ServiceNow to SL1" either manually or on a schedule to create and execute the discovery session in SL1.
6. After the discovery session has completed in SL1, run "Sync Discovery Session Status from SL1 to ServiceNow" either manually or on a schedule to update the status of the RITM in ServiceNow.

NOTE: The following integration applications only sync CIs that are aligned with the devices that are already synced with ServiceNow. Before syncing any of the CIs below, you must first sync devices between SL1 and ServiceNow.

Sync Business Services from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync Business Services from SL1 to ServiceNow
2. Configure the service classification mappings in "Sync Business Services from SL1 to ServiceNow". These are defined in the **Configuration** pane.
3. Run "Sync Business Services from SL1 to ServiceNow" either manually or on a schedule.

Sync File Systems from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync File Systems from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs.
3. Run "Sync File Systems from SL1 to ServiceNow" either manually or on a schedule.

Sync Network Interfaces from SL1 to ServiceNow

1. Align the configuration object to the following integration application:

- Sync Network Interfaces from SL1 to ServiceNow
2. Determine additional filters for syncing network interfaces using the **adapter_sync** variable defined in the **Configuration** pane.
 3. The parent CI must be synced in order to see these related CIs.
 4. Run "Sync Network Interfaces from SL1 to ServiceNow" either manually or on a schedule. This integration application should be run at least every 23 hours if you would like to sync interface-level relationships with "Sync Advanced Topology from SL1 to ServiceNow".

Sync Installed Software from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync Software Packages from SL1 to ServiceNow
 - Sync Installed Software from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs.
3. First, run "Sync Software Packages from SL1 to ServiceNow". Verify that the integration has run successfully.
4. Run "Sync Installed Software from SL1 to ServiceNow" either manually or on a schedule.

Sync Maintenance Schedules from ServiceNow to SL1

1. Align the configuration object to the following integration application:
 - Sync Maintenance Schedules from ServiceNow to SL1
2. The parent CI must be synced in order to see these related CIs. The affected CI must have **SL1 Monitored** set to *True*.
3. If needed, configure the **Request** and **Task state** IDs in the **Configuration** pane. This is needed if the customer has custom IDs for certain change request or change task states.
4. Run "Sync Maintenance Schedules from ServiceNow to SL1" either manually or on a schedule.

Sync Advanced Topology from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync Advanced Topology from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs. Syncing Interface-level relationships (Layer 2, LLDP, and CDP) will require that "Sync Network Interfaces from SL1 to ServiceNow" is set to run at least every 23 hours.
3. Run "Sync Advanced Topology from SL1 to ServiceNow" either manually or on a schedule.

CMDB-Only ServiceNow Integration with Single SL1 and Domain-Separated ServiceNow

1. Deploy the Integration Service ISO:
 - IP address, Netmask, Gateway, DNS, Hostname provided
 - Root password provided (this is the root user for the OS)
 - Start Docker services after installation:

```
/opt/iservices/scripts/pull_start_iservices.sh
```
 - Validate that iservices are running:

```
docker service ls
```
2. Activate the Configuration Management For Scoped Apps (CMDB) Plugin.
3. Install the ScienceLogic Certified Application and create a ServiceNow group and user account:
 - Username
 - Password
 - Web Service Access Only
 - GMT Time Zone
 - x_sclo_scilogic.Admin role assigned
4. Install the ServiceNow SyncPacks on the Integration Service.
5. Create the Integration Service configuration object using the "ServiceNow SyncPack" configuration object as a template.

Sync Devices from SL1 to ServiceNow

1. Align the configuration object to following integration applications:
 - Sync Organizations from SL1 to ServiceNow Companies
 - Cache ServiceNow CIs and SL1 Device Classes
 - Sync Devices from SL1 to ServiceNow
2. In "Sync Organizations from SL1 to ServiceNow Companies" **Configuration** pane, set the **Source_of_Truth** to *ServiceNow* and set the **Domain_Separation** flag to *True*.
3. Run "Sync Organizations from SL1 to ServiceNow Companies".
4. Run "Cache ServiceNow CIs and SL1 Device Classes" to retrieve all device class information from SL1 and ServiceNow. This will populate the device class mapping in the following step. This integration application should be run at least every 23 hours.

5. Configure class and attribute mappings in "Sync Devices from SL1 to ServiceNow". Set **Domain_Separation** to *True*.
6. Run "Generate Required CI Relations for ServiceNow" to see if you are missing any service rules or class mappings, and then create any required maps, containment rules, and hosting rules.
7. Run "Sync Devices from SL1 to ServiceNow" either manually or on a schedule. This integration application should be run at least every 23 hours.

Discover Devices from ServiceNow in SL1

1. Align Configuration Object to following integration applications:
 - Sync Discovery Requirements
 - Sync Discovery Session Requests from ServiceNow to SL1
 - Sync Discovery Session Status from SL1 to ServiceNow
2. In "Sync Discovery Requirements" **Configuration** pane, set the **Source_of_Truth** to *ServiceNow*. Set **Domain_Separation** to *True*.
3. Run "Sync Discovery Requirements" to sync all discovery-dependent information from SL1 to ServiceNow and back to SL1.
4. Set additional configuration variables for each of the other integration applications above in the respective **Configuration** pane.
5. Discovery request RITM is successfully created and approved in ServiceNow using the provided Service Catalogs.
6. Run "Sync Discovery Session Requests from ServiceNow to SL1" either manually or on a schedule to create and execute the discovery session in SL1.
7. After the discovery session completes in SL1, run "Sync Discovery Session Status from SL1 to ServiceNow" either manually or on a schedule to update the status of the RITM in ServiceNow.

NOTE: Domain separation requires that "Sync Organizations from SL1 to ServiceNow Companies" is configured with **Domain_Separation** enabled and **Source_of_Truth** set to *ServiceNow*. In a domain-separated ServiceNow environment, this integration application must be properly configured and run successfully before syncing any additional CI items. SL1 organizations that are linked to a ServiceNow company will have the **crm_id** populated with the ServiceNow Company **sys_id**.

NOTE: The following integrations only sync CIs that are aligned with the devices that are already synced with ServiceNow. Before syncing any of the CIs below, you must first sync devices between SL1 and ServiceNow.

Sync Business Services from SL1 to ServiceNow

1. Align the configuration object to the following integration application:

- Sync Business Services from SL1 to ServiceNow
2. Configure the service classification mappings in "Sync Business Services from SL1 to ServiceNow". These are defined in the **Configuration** pane.
 3. Ensure that **Domain_Separation** is set to *True* in the **Configuration** pane.
 4. Run "Sync Business Services from SL1 to ServiceNow" either manually or on a schedule.

Sync File Systems from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync File Systems from SL1 to ServiceNow
2. Ensure that **Domain_Separation** is set to *True* in the **Configuration** pane.
3. The parent CI must be synced in order to see these related CIs.
4. Run "Sync File Systems from SL1 to ServiceNow" either manually or on a schedule.

Sync Network Interfaces from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync Network Interfaces from SL1 to ServiceNow
2. Determine additional filters for syncing network interfaces using the **adapter_sync** variable defined in the **Configuration** pane.
3. Ensure that **Domain_Separation** is set to *True* in the **Configuration** pane.
4. The parent CI must be synced in order to see these related CIs.
5. Run "Sync Network Interfaces from SL1 to ServiceNow" either manually or on a schedule. This integration application should be run at least every 23 hours if you would like to sync interface-level relationships with "Sync Advanced Topology from SL1 to ServiceNow".

Sync Installed Software from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync Software Packages from SL1 to ServiceNow
 - Sync Installed Software from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs.
3. First, run "Sync Software Packages from SL1 to ServiceNow". Verify that the integration has run successfully.
4. Run "Sync Installed Software from SL1 to ServiceNow" either manually or on a schedule.

Sync Maintenance Schedules from ServiceNow to SL1

1. Align the configuration object to the following integration application:

- Sync Maintenance Schedules from ServiceNow to SL1
2. The parent CI must be synced in order to see these related CIs. The affected CI must have **SL1 Monitored** set to *True*.
 3. If needed, configure the **Request** and **Task state** IDs in the **Configuration** pane. This is needed if the customer has custom IDs for certain change request or change task states.
 4. Run "Sync Maintenance Schedules from ServiceNow to SL1" either manually or on a schedule.

Sync Advanced Topology from SL1 to ServiceNow

1. Align the configuration object to the following integration application:
 - Sync Advanced Topology from SL1 to ServiceNow
2. Ensure that **Domain_Separation** is set to *True* in the **Configuration** pane.
3. The parent CI must be synced in order to see these related CIs. Syncing Interface-level relationships (Layer 2, LLDP, and CDP) will require that "Sync Network Interfaces from SL1 to ServiceNow" is set to run at least every 23 hours.
4. Run "Sync Advanced Topology from SL1 to ServiceNow" either manually or on a schedule.

CMDB-Only ServiceNow Integration with Multiple SL1 Systems, no Domain-Separated ServiceNow

NOTE: Depending on the size of your SL1 stacks and the number of SL1 stacks you have, you may need to consider a “multi-tenant” configured IS. This is a more advanced deployment model. Please contact a ScienceLogic representative for more information.

1. Deploy the Integration Service ISO:

- IP address, Netmask, Gateway, DNS, Hostname provided
- Root password provided (this is the root user for the OS)
- Start Docker services after installation:

```
/opt/iservices/scripts/pull_start_iservices.sh
```

- Validate that iservices are running:

```
docker service ls
```

2. Activate the Configuration Management For Scoped Apps (CMDB) Plugin.

3. Install the ScienceLogic Certified Application and create a ServiceNow group and user account:

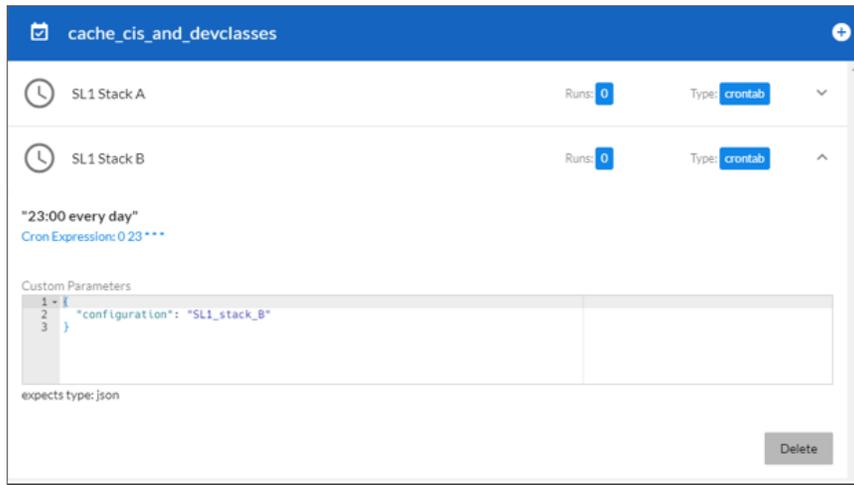
- Username
- Password
- Web Service Access Only
- GMT Time Zone
- x_sclo_scilogic.Admin role assigned

4. Install the ServiceNow SyncPacks on the Integration Service.

5. Create the Integration Service configuration object using the "ServiceNow SyncPack" configuration object as a template:

NOTE: The key difference between integrating a single SL1 stack on the Integration Service and integrating multiple SL1 stacks on the Integration Service is how you run the integration application. Running the integration application with multiple SL1 stacks involves creating an individual configuration object for each SL1 stack. Then, create an individual schedule for each configuration object. Each schedule should use a configuration object that is specific to a single SL1 stack.

When creating the schedule, populate the custom parameters with the configuration object ID. For example:



Sync Devices from SL1 to ServiceNow

1. Create a schedule for each SL1 stack for the following Integrations following the note above:
 - o Cache ServiceNow CIs and SL1 Device Classes
 - o Sync Devices from SL1 to ServiceNow
2. Wait for the "Cache ServiceNow CIs and SL1 Device Classes" to retrieve all device class information from SL1 and ServiceNow. This will populate the device class mapping in the following step. This integration should be run at least every 23 hours.
3. Configure class and attribute mappings in "Sync Devices from SL1 to ServiceNow". If each SL1 stack has different class and attribute mapping requirements, you will need to specify the mappings in each schedule's custom parameters for this application.
4. Run "Generate Required CI Relations for ServiceNow" to see if you are missing any service rules or class mappings and create any required maps, containment rules, and hosting rules. By default, these will pull from the mappings set in "Sync Devices from SL1 to ServiceNow". If you have defined different mappings for each SL1 stack, you will need to also specify these mappings in each schedule's custom parameters for this application.
5. Run "Sync Devices from SL1 to ServiceNow" with a schedule where each schedule uses the configuration object for a SL1 stack. This integration application should be run at least every 23 hours.

Discover Devices from ServiceNow in SL1

1. Create a schedule for each SL1 stack for the following integration applications:
 - o Sync Discovery Requirements
 - o Sync Discovery Session Requests from ServiceNow to SL1

- Sync Discovery Session Status from SL1 to ServiceNow
2. If any additional configuration variables are needed, these will need to be set in the custom parameters section for each schedule
 3. Run "Sync Discovery Requirements" on a schedule to sync all discovery-dependent information from SL1 to ServiceNow.
 4. Verify that the discovery request RITM is successfully created and approved in ServiceNow using the provided Service Catalogs.
 5. Run "Sync Discovery Session Requests from ServiceNow to SL1" on a schedule where each schedule uses the configuration object for a SL1 stack to create and execute the discovery session in SL1.
 6. Once the discovery session has completed in SL1, run "Sync Discovery Session Status from SL1 to ServiceNow" via schedule where each schedule uses the configuration object for a SL1 stack to update the status of the RITM in ServiceNow.

The following integration applications only sync CIs that are aligned with the devices that are already synced with ServiceNow. Before syncing any of the CIs below, you must first sync devices between SL1 and ServiceNow.

Sync Business Services from SL1 to ServiceNow

1. Create a schedule for each SL1 stack for the following integration application:
 - Sync Business Services from SL1 to ServiceNow
2. Configure service classification mappings in "Sync Business Services from SL1 to ServiceNow" application. These are defined in the **Configuration** pane.
3. Run "Sync Business Services from SL1 to ServiceNow" on a schedule where each schedule uses the configuration object for a SL1 stack.

Sync File Systems from SL1 to ServiceNow

1. Create a schedule for each SL1 stack for the following integration application:
 - Sync File Systems from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs.
3. Run "Sync File Systems from SL1 to ServiceNow" on a schedule where each schedule uses the configuration object for a SL1 stack.

Sync Network Interfaces from SL1 to ServiceNow

1. Create a schedule for each SL1 stack for the following integration application:
 - Sync Network Interfaces from SL1 to ServiceNow

2. Determine additional filters for syncing network interfaces using the **adapter_sync** variable defined in the **Configuration** pane.
3. The parent CI must be synced in order to see these related CIs.
4. Run "Sync Network Interfaces from SL1 to ServiceNow" on a schedule where each schedule uses the configuration object for a SL1 stack. This integration application should be run at least every 23 hours if you would like to sync interface-level relationships with "Sync Advanced Topology from SL1 to ServiceNow".

Sync Installed Software from SL1 to ServiceNow

1. Create a schedule for each SL1 stack for the following integration application:
 - Sync Software Packages from SL1 to ServiceNow
 - Sync Installed Software from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs.
3. "Sync Software Packages from SL1 to ServiceNow" must run on a schedule before "Sync Installed Software from SL1 to ServiceNow". Verify that the software packages have been synced before continuing.
4. After the software packages have been synced, run "Sync Installed Software from SL1 to ServiceNow" on a schedule where each schedule uses the configuration object for a SL1 stack.

Sync Maintenance Schedules from ServiceNow to SL1

1. Create a schedule for each SL1 stack for the following integration application:
 - Sync Maintenance Schedules from ServiceNow to SL1
2. The parent CI must be synced in order to see these related CIs. The affected CI must have the **SL1 Monitored** field set to *True*.
3. If needed, configure the **Request** and **Task state** IDs in the **Configuration** pane. This is needed if the customer has custom IDs for certain change request or change task states.
4. Run "Sync Maintenance Schedules from ServiceNow to SL1" on a schedule where each schedule uses the configuration object for a SL1 stack.

Sync Advanced Topology from SL1 to ServiceNow

1. Create a schedule for each SL1 stack for the following integration application:
 - Sync Advanced Topology from SL1 to ServiceNow
2. The parent CI must be synced in order to see these related CIs. Syncing Interface-level relationships (Layer 2, LLDP, and CDP) will require that you set "Sync Network Interfaces from SL1 to ServiceNow" to run at least every 23 hours.
3. Run "Sync Advanced Topology from SL1 to ServiceNow" on a schedule where each schedule uses the configuration object for a SL1 stack.

Incident-Only ServiceNow Integration with Single SL1, no Domain Separation in ServiceNow

1. Deploy the Integration Service ISO:
 - IP address, Netmask, Gateway, DNS, Hostname provided
 - Root password provided (this is the root user for the OS)
 - Start Docker services after installation:

```
/opt/iservices/scripts/pull_start_iservices.sh
```
 - Validate that iservices are running:

```
docker service ls
```
2. Install the ScienceLogic Certified Application and create a ServiceNow group and user account:
 - Username
 - Password
 - Web Service Access Only
 - GMT Time Zone
3. Install the ServiceNow SyncPacks on the Integration Service.
4. Create the Integration Service configuration object using the "ServiceNow SyncPack" configuration object as a template:
 - Align the configuration object to the following integration applications:
 - Create or Update ServiceNow Incident from SL1 Event
 - Update ServiceNow Incident when SL1 Event is Acknowledged
 - Update ServiceNow Incident when SL1 Event is Cleared
 - Sync Incident State from ServiceNow to SL1 Event
 - Only the following Integration should be run manually or scheduled:
 - Sync Incident State from ServiceNow to SL1 Event
5. Install the ServiceNow Base PowerPack and configure SL1 :
 - Use the "ServiceNow RBA – Example" credential as a template to create a new credential that points to the Integration Service instance.
 - Align the newly-created credential to the "ServiceNow – Add/Update/Clear Incident" Run Book Action.
 - Ensure that all Run Book Actions and Run Book Policies are enabled.

Appendix

B

Certified Application Objects

Overview

This appendix describes the tables, endpoints, and roles that were created in ServiceNow as part of the "ScienceLogic SL1 : CMDB & Incident Automation" application. This application is also known as the "Certified Application" or the "Scoped Application".

This appendix includes the following topics:

<i>Roles</i>	164
<i>Tables</i>	165
<i>Table Columns (cmdb_ci)</i>	165
<i>Table Columns (core_company)</i>	166
<i>Table Columns (cmdb_group)</i>	166
<i>Script Includes</i>	166
<i>Event Registry</i>	166
<i>Scripted Actions</i>	167
<i>Data Lookup Definitions</i>	167
<i>System Properties</i>	167
<i>Catalog Item</i>	168
<i>Catalog UI Policies</i>	168
<i>Variable Sets</i>	168
<i>Catalog Client Scripts</i>	169
<i>Workflows</i>	169
<i>Scripted REST Resources</i>	170

Roles

Two Roles were added with the ScienceLogic update set, Admin (x_sclo_scilogic.Admin) and User (x_sclo_scilogic.User). Both give access to SL1.

Role	Inherited Roles	Other Inherited Roles	Role Definition
x_sclo_scilogic.Admin			Role for ScienceLogic Service Accounts.
	itil		Can perform standard actions for an ITIL help desk technician. This is the default "Technician" role. Can open, update, close incidents, problems, changes, config management items. By default, only users with the itil role can have tasks assigned to them
		Dependency_view	A special role to be applied both on the \$ngbsm UI page and on the BSMPProcessor. This role is required to access the dependency views module. By default, ITIL includes this role to avoid regressions.
		cmdb_query_builder	Can access the CMDB Query Builder application to create, run, and save queries on the CMDB.
		template_editor	
		view_changer	Can switch active views.
		app_service_user	Can view and retrieve information using API from application service maps (cmdb_ci_service_discovered).
		certification	Can work on Certification tasks.
	import_transformer		Can manage Import Set Transform Maps and run transforms.
x_sclo_scilogic.User			General user account that allows read-only access to SL1.

Tables

Name	Label	Extends	Comments
x_sclo_scilogic_event	Event	(empty)	Event information
x_sclo_scilogic_event_severity	Event Severity Look Rules	Data Lookup Matcher Rules	Look up table for event Severity
x_sclo_scilogic_incident	Import Incident	Import Set Row	Import / staging events before transform to Event and Incident
x_sclo_scilogic_import_installed_software	Import Installed Software	Import Set Row	Import / staging events before transform to Software Instance
x_sclo_scilogic_org_ven_mfg	Import ORG VEN MFG	Import Set Row	Import / staging events before transform to core_company
x_sclo_scilogic_import_discovery_dependent	Import Discovery Dependent	Import Set Row	Import / staging events before transform to Discovery Dependent table
x_sclo_scilogic_discovery_dependent	Discovery Dependent	(empty)	Discovery Dependent Information
x_sclo_scilogic_catalog_item_templates	Catalog item Templates	(empty)	Templates use to fill out catalog items
x_sclo_scilogic_import_service_request	Import Service Request	Import Set Row	Import / staging events before transform to Service Requests

Table Columns (cmdb_ci)

Name	Label	Type	Comments
x_sclo_scilogic_id	SL1 ID	Integer	Unique ID
x_sclo_scilogic_region	SL1 Region	String	Unique String of SL1 Platform
x_sclo_scilogic_url	SL1 URL	URL	URL to SL1 Platform
x_sclo_scilogic_monitored	SL1 Monitored	True/False	Device currently synced with SL1 Platform

Table Columns (core_company)

Name	Label	Type	Comments
x_sclo_scilogic_id	SL1 ID	String	Unique ID
x_sclo_scilogic_region	SL1 Region	String	Unique String of SL1 Platform
x_sclo_scilogic_monitored	SL1 Monitored	True/False	Organization currently synced with SL1 Platform

Table Columns (cmdb_group)

Name	Label	Type	Comments
x_sclo_scilogic_id	SL1 ID	String	Unique ID
x_sclo_scilogic_region	SL1 Region	String	Unique String of SL1 Platform

Script Includes

Name	API Name	Comments
CatalogUtils	x_sclo_scilogic.catalogUtils	Catalog Script include scripts
ChangeUtils	x_sclo_scilogic.changeUtils	Change Script include scripts
DeviceUtils	x_sclo_scilogic.DeviceUtils	Device Script include scripts
EventUtils	x_sclo_scilogic.EventUtils	Event Script include scripts
GeneralUtils	x_sclo_scilogic.GeneralUtils	General Script include scripts

Event Registry

Suffix	Event name	Table	Comments
device_monitoring	x_sclo_scilogic.device_monitoring	Configuration Item [cmdb_ci]	Event for Device Monitoring
Remove_monitoring	x_sclo_scilogic.remove_monitoring	Configuration Item [cmdb_ci]	Event for Remove Monitoring

Scripted Actions

Name	Event name	Comments
Device Monitoring Catalog item	x_sclo_scilogic.device_monitoring	Action used to submit Catalog item via Event.
Device Removal Catalog item	x_sclo_scilogic.remove_monitoring	Action used to submit Catalog item via Event.

Data Lookup Definitions

Name	Source Table	Matcher Table	Comments
Event Severity	Import Incident [x_sclo_scilogic_incident]	Event Severity Lookup Rules [x_sclo_scilogic_event_severity]	Lookup for ScienceLogic Severity to Impact and Urgency

System Properties

Suffix	Name	Comments
CatalogItemDiscovery	x_sclo_scilogic.CatalogItemDiscovery	Unique value (sys_id)
CatalogItemRemove	x_sclo_scilogic.CatalogItemRemove	Unique value (sys_id)
closeCode	x_sclo_scilogic.closeCode	Value to use for Close Code for Incident Transform
Contact type	x_sclo_scilogic.Contact Type	Value to use for Contact type for Incident Transform
deviceLogging	x_sclo_scilogic.deviceLogging	Turn on Logging
deviceLoggingParam	x_sclo_scilogic.deviceLoggingParm	Add additional parameters beyond the default errors
discoverySource	x_sclo_scilogic.discoverySource	Discovery Source to be used by Integration Service
notResolved	x_sclo_scilogic.notResolved	Value of Reopened Incident
stateNew	x_sclo_scilogic.stateNew	Value of New Incident
StateResolved	x_sclo_scilogic.stateResolved	Value of Resolved Incident

Catalog Item

Name	Comments
Device Discovery	Role for ScienceLogic Service Accounts.
Monitoring Removal	General user account that allows read only access to ScienceLogic Application.

Catalog UI Policies

Catalog item	Short description	Comments
Device Discovery	Catalog Template	Updates form based on Select template
Device Discovery	Create Virtual Device	Updates form based on Request type
Device Discovery	Create Virtual Device (Retired)	
Device Discovery	Device Discovery	Updates form based on Request type
Device Discovery	Device Discovery (Retired)	
Monitoring Removal	Hide Overview variables not required	Hide variables not required for the Monitoring Removal request
Device Discovery	Port Scan	Hide scan ports that are not default
Device Discovery	Port Scan (Retired)	
Device Discovery	Region	Updates form based on Organization
Device Discovery	Region (Retired)	
Monitoring Removal	Region via Organization	Updates form based on Organization
Device Discovery	Save as Template	Updates form based on Save as template

Variable Sets

Title	Internal name	Comments
Create_virtual_device	create_virtual_device	
Discovery Overview	discovery_overview	

Title	Internal name	Comments
Discovery Sesion - Basic Settings	discovery_sesion_basic_settings	
Discovery Session - Detection and Scanning	discovery_session_detection_and_scanning	
Discovery Session - IP & Credentials	discovery_session_ip_credentials	
Monitoring Removal	monitoring_removal	
Service Catalog item Template	service_catalog_item_template	

Catalog Client Scripts

Name	Catalog item	Type	Comments
Hide Request Type Options	Monitoring Removal	onLoad	Shared variable hide options that don't apply
Hide Request Type Options	Device Discovery	onLoad	Shared variable hide options that don't apply
Region	Monitoring Removal	onChange	Update Region field based on Company Region
Region	Monitoring Removal	onChange	Update Region field based on Company Region

Workflows

Name	Table	Comments
SL1 Monitoring Removal	Requested Item [sc_req_item]	Workflow for Removal of devices from SL1 process
SL1 Discovery Session	Requested Item [sc_req_item]	Workflow for Discovery session process

Scripted REST Resources

Name		Comments	
Business Services	/api/x_sclo_scilogic/v1/sciencelogic/business_service	GET	This GET api will pull all ScienceLogic monitored Configuration items specific to Business Services class from the CMDB. It will be ordered via the sys_id field to ensure the same order every time.
CMDB Group	/api/x_sclo_scilogic/v1/sciencelogic/cmdb_group	POST	Use this API to create cmdb_groups & add a CI to them.
Change Requests	/api/x_sclo_scilogic/v1/sciencelogic/change_requests	GET	This GET api will pull Active Change Requests or Change Tasks based on the record_type supplied that have ScienceLogic monitored CI attached. It will be ordered via the sys_id field to ensure the same order every time.
Classification	/api/x_sclo_scilogic/v1/sciencelogic/classification	GET	This GET api will pull all required CMDB information to build JSON payloads.
Companies	/api/x_sclo_scilogic/v1/sciencelogic/companies	GET	This GET api will pull all Active Companies that are ScienceLogic monitored. It will be ordered via the sys_id field to ensure the same order every time.
Configuration Items	/api/x_sclo_scilogic/v1/sciencelogic/configuration_Items	GET	This GET api will pull all ScienceLogic monitored Configuration items from the CMDB. It will be ordered via the sys_id field to ensure the same order every time.
Device IdentificationEngine	/api/x_sclo_scilogic/v1/sciencelogic/IdentificationEngine	POST	Use this API to create or update configuration items within the CMDB via ScienceLogic.

Name		Comments	
File Systems	/api/x_sclo_scilogic/v1/sciencelogic/file_systems	GET	This GET api will pull all ScienceLogic monitored Configuration items specific to File systems class from the CMDB. It will be ordered via the sys_id field to ensure the same order every time.
Import Set	/api/x_sclo_scilogic/v1/sciencelogic/import_set	POST	This POST API will post to the target import set table and create a record for each cmdb_ci.
Incidents	/api/x_sclo_scilogic/v1/sciencelogic/incidents	GET	This GET api will pull all incidents. It will be ordered via the sys_id field to ensure the same order every time.
Installed Software	/api/x_sclo_scilogic/v1/sciencelogic/installed_software	GET	This GET api will pull all Servicenow Software packages and installed instances from the CMDB. It will be ordered via the sys_id field to ensure the same order every time.
Manufacture	/api/x_sclo_scilogic/v1/sciencelogic/manufactures	POST	This POST API will pull all Manufactures.
Model	/api/x_sclo_scilogic/v1/sciencelogic/models	POST	This POST API will pull all Model.
Network Adapters	/api/x_sclo_scilogic/v1/sciencelogic/network_adapters	GET	This GET api will pull all ScienceLogic monitored Configuration items specific to Network Adapter class from the CMDB. It will be ordered via the sys_id field to ensure the same order every time.
Service Request	/api/x_sclo_scilogic/v1/sciencelogic/service_request	GET	This GET api will pull all ServiceRequest items from the CMDB associated with Device Discovery Catalog item. It will be ordered via the sys_id field to ensure the same order every time.
Classification	/api/x_sclo_scilogic/v2/sciencelogic/classification	GET	This GET api will pull all required CMDB information to build JSON payloads.

Transform Maps

Name	Source Table	Target Table	Comments
ScienceLogic Discovery Dependent	Import Discovery Dependent	Discovery Dependent	Import / staging table for Catalog Dependents
ScienceLogic Event	Import Incident	Event	Import / staging table for Events.
ScienceLogic Incident	Import Incident	Incident [incident]	Import / staging table for Incident
ScienceLogic Organization	Import ORG VEN MFG []	Company [core_company]	Import / staging table for Organization
ScienceLogic Service Request	Import Service Request []	Request Item [sc_req_item]	Import / staging table for Request item

Transform Scripts

Name	Transform Map	Order	Comments
onBefore	ScienceLogic Event	100	Check Action
onAfter	ScienceLogic Event	100	Check Action; Get Resolved Validation script include
onBefore	ScienceLogic Incident	100	Check Action, event workflow script include
onAfter	ScienceLogic Incident	100	Check Action, Affected CI script include

Appendix

C

ServiceNow API Endpoints

Overview

This appendix describes the customized ServiceNow API Endpoints that were created for the Integration Service ServiceNow SyncPacks. These scripted endpoints reduce the amount of REST calls that the Integration Service makes to ServiceNow.

Please note that for pagination, the following Query parameters are not required: `sysparm_offset` and `sysparm_limit`. The default settings are:

- `sysparm_offset=0`
- `sysparm_limit` = ServiceNow defines the default upper limits for data export. It will check the following properties at *System Properties > Import Export*: `glide.json.export.limit`, `glide.ui.export.limit`, and then `glide.ui.export.war.threshold`.

For example, if you have 200 total records and you want to pull the records in 100-record chunks, then the first pull would be `sysparm_offset=0` & `sysparm_limit=100` and the second pull would be `sysparm_offset=100` & `sysparm_limit=100`. For more information, see the ServiceNow documentation for [Export Limits](#).

This appendix includes the following topics:

Business Services	175
Change Requests	177
Classification version 1	179
Classification version 2	182
CMDB Group	185
Companies	187
Configuration Items	189

<i>Device Identification Engine</i>	191
<i>File Systems</i>	194
<i>Import Set</i>	196
<i>Incidents</i>	198
<i>Installed Software</i>	200
<i>Manufacturer</i>	203
<i>Model</i>	205
<i>Network Adapters</i>	207
<i>Service Requests</i>	209

Business Services

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/business_service`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/business_service`

This operation pulls all the fields from just the Business Service (**cmdb_ci_service**) table. The return is ordered by **sys_id**, so the results display in the same order every time. The results are filtered by the **SL1 monitored** and **SL1 ID** field on the ServiceNow side. This operation requires the region to be supplied by the requester, and it will only return region-supplied configuration items.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

Example (Request URL)

`https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/ sciencelogic/business_service`

Example (Response)

```
{
  "results": [
    {
      "operational_status": "1",
      "sys_updated_on": "2019-02-06 19:32:34",
      "discovery_source": "Other Automated",
      "first_discovered": "2019-02-06 19:31:19",
      "sys_updated_by": "admin",
      "sys_created_on": "2019-02-06 19:31:19",
      "sys_domain": "global",
      "used_for": "Production",
      "sys_created_by": "is4user1",
      "sys_domain_path": "/",
      "install_status": "1",
      "name": "One Service to rule them",
      "subcategory": "Service",
      "busines_criticality": "1 - most critical",
      "last_discovered": "2019-02-06 19:31:19",
      "sys_class_name": "cmdb_ci_service",
      "sys_id": "52da95dcdb6323009f7dd7a0cf961918",
      "sys_class_path": "/!/#C",
      "comments": "Postman",
      "sys_mod_count": "1",
      "x_sclo_scilogic_id": "1570",
      "model_id": "e8aaeb3f3763100044e0bfc8bcbe5d20",
      "cost_cc": "USD",
      "x_sclo_scilogic_monitored": "true",
      "category": "Business Service",
      "service_classification": "Technical Service",
      "x_sclo_scilogic_region": "ScienceLogic"
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 100,
  "return_count": 1,
  "total_count": 1
}
```

Change Requests

HTTP Method

GET

Resource Path

```
/api/x_sclo_scilogic/v1/sciencelogic/change_requests?record_type=change_request&state=1&region=ScienceLogic
```

Default Resource Path

```
/api/x_sclo_scilogic/v1/sciencelogic/change_requests?record_type=change_request&state=1&region=ScienceLogic
```

This scripted API was built for pulling Change Requests or Change Tasks and formatting a JSON object response with the required information to create a maintenance schedule in SL1. The GET queries the **task_ci** table to find configuration items that are monitored by SL1 and are the correct record type. The GET operation returns all records with their configuration items in formatted JSON strings that include planned start and end time.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
record_type (required)	change_request
state	-5
region (required)	ScienceLogic
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status

Code	Value
200	OK
400	Query parameter \region\ is not defined and is required.

Fixed Internal Query

State:

```
`task.sys_class_name=' + recordType + `task.state=' + state + `^ci_item.x_sclo_scilogic_monitored=true^ci_item.x_sclo_scilogic_region=' + region
```

Non-State:

```
`task.active=true^task.sys_class_name=' + recordType + `ci_item.x_sclo_scilogic_monitored=true^ci_item.x_sclo_scilogic_region=' + region
```

Example

https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/change_requests?record_type=change_request&state=-5®ion=ScienceLogic

Example (Response)

```
{
  "results": [
    {
      "sys_id": "48ebaba0db962f00dc44f00fbf961961",
      "number": "CHG0030001",
      "state_value": "-5",
      "state": "New",
      "short_description": "Test Change",
      "planned_start_date": "2019-01-01 06:00:01",
      "planned_end_date": "2019-01-01 18:00:01",
      "device": [
        {
          "sys_id": "d83dac0adb4dab00dc44f00fbf961919",
          "name": "Postman Test Server 11",
          "id": "11",
          "region": "ScienceLogic"
        }
      ]
    }
  ]
}
```

Classification version 1

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/classification`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/classification`

To support the identification and reconciliation framework, SL1 requires a large amount of information to know how to correctly fill out the JSON formatted string defined by the Identification Engine documentation. This operation uses the **getTableExtension()** function to find all the tables extended from the **cmdb_ci** table and then goes through each table one by one. This operation collects information about each class, such as which fields are required to identify and if it considers another class to help find uniqueness. This operation then finds all the associated metadata. Finally, the operation pulls a list of all field names from the table. By default the **criterion_attributes** and **attributes** are not included and require "action=attributes" as a parameter in the API call to be passed.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
action	attributes

Attributes require x_sclo_scilogic.Admin be added to **sys_dictionary.*** (read) ACL to allow the API to access field names on each class table.

Fixed Internal Query

Example

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/sciencelogic/
classification
```

Example (Response)

```
{
  "results": [
    {
      "class_label": "Storage Area Network",
      "class_table": "cmdb_ci_san",
      "criterion_attributes": [
        ""
      ],
      "independent": "false",
      "containment_rule": [

    ],
    "hosting_rule": [

  ],
  "reference_rule": [

],
  "attributes": [
    "asset",
    "asset_tag",
    "assigned",
    "assigned_to",
    "assignment_group",
    "attributes",
    "can_print",
    "category",
    "change_control",
    "checked_in",
    "checked_out",
    "comments",
    "company",
    "correlation_id",
    "cost",
    "cost_cc",
    "cost_center",
    "delivery_date",
    "department",
    "discovery_source",
    "dns_domain",
    "due",
    "due_in",
    "fault_count",
    "first_discovered",
    "fqdn",
    "gl_account",
    "install_date",
    "install_status",
    "invoice_number",
```

```
"ip_address",
"justification",
"last_discovered",
"lease_id",
"location",
"mac_address",
"maintenance_schedule",
"managed_by",
"manufacturer",
"model_id",
"model_number",
"monitor",
"name",
"operational_status",
"order_date",
"owned_by",
"po_number",
"purchase_date",
"san_id",
"schedule",
"serial_number",
"short_description",
"skip_sync",
"start_date",
"subcategory",
"supported_by",
"support_group",
"sys_class_name",
"sys_class_path",
"sys_created_by",
"sys_created_on",
"sys_domain",
"sys_domain_path",
"sys_id",
"sys_mod_count",
"sys_updated_by",
"sys_updated_on",
"unverified",
"vendor",
"warranty_expiration",
"x_sclo_scilogic_id",
"x_sclo_scilogic_monitored",
"x_sclo_scilogic_region",
"x_sclo_scilogic_url"
]
}
]
```

Classification version 2

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v2/sciencelogic/classification`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/classification`

To support the identification and reconciliation framework, SL1 requires a large amount of information to know how to correctly fill out the JSON formatted string defined by the Identification Engine documentation. This operation uses the **getTableExtension()** function to find all the tables extended from the **cmdb_ci** table and then goes through each table one by one. This operation collects information about each class, such as which fields are required to identify and if it considers another class to help find uniqueness. This operation then finds all the associated metadata. Finally, the operation pulls a list of all field names from the table. By default the **criterion_attributes** and **attributes** are not included and require "action=attributes" as a parameter in the API call to be passed.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
action	attributes
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

Attributes require x_sclo_scilogic.Admin be added to **sys_dictionary.*** (read) ACL to allow the API to access field names on each class table.

Fixed Internal Query

Example

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v2/sciencelogic/
classification
```

Example (Response)

```
{
  "results": [
    {
      "class_label": "Storage Area Network",
      "class_table": "cmdb_ci_san",
      "criterion_attributes": [
        ""
      ],
      "independent": "false",
      "containment_rule": [

    ],
    "hosting_rule": [

  ],
  "reference_rule": [

],
  "attributes": [
    "asset",
    "asset_tag",
    "assigned",
    "assigned_to",
    "assignment_group",
    "attributes",
    "can_print",
    "category",
    "change_control",
    "checked_in",
    "checked_out",
    "comments",
    "company",
    "correlation_id",
    "cost",
    "cost_cc",
    "cost_center",
    "delivery_date",
    "department",
    "discovery_source",
    "dns_domain",
    "due",
    "due_in",
    "fault_count",
    "first_discovered",
    "fqdn",
    "gl_account",
    "install_date",
    "install_status",
    "invoice_number",
```

```
"ip_address",
"justification",
"last_discovered",
"lease_id",
"location",
"mac_address",
"maintenance_schedule",
"managed_by",
"manufacturer",
"model_id",
"model_number",
"monitor",
"name",
"operational_status",
"order_date",
"owned_by",
"po_number",
"purchase_date",
"san_id",
"schedule",
"serial_number",
"short_description",
"skip_sync",
"start_date",
"subcategory",
"supported_by",
"support_group",
"sys_class_name",
"sys_class_path",
"sys_created_by",
"sys_created_on",
"sys_domain",
"sys_domain_path",
"sys_id",
"sys_mod_count",
"sys_updated_by",
"sys_updated_on",
"unverified",
"vendor",
"warranty_expiration",
"x_sclo_scilogic_id",
"x_sclo_scilogic_monitored",
"x_sclo_scilogic_region",
"x_sclo_scilogic_url"
]
}
]
```

CMDB Group

HTTP Method

POST

Pagination

Enabled

Resource Path

```
/api/x_sclo_scilogic/v1/sciencelogic/cmdb_group
```

Default Resource Path

```
/api/x_sclo_scilogic/sciencelogic/cmdb_group
```

This operation handles the intake of groups of devices from SL1 and converts the device groups to CMDB groups. This operation uses a standard formatted JSON string, and it checks for a **sys_id** of the group first by searching for a matching group. This process creates a group if a group is not supplied or found, and then it passes the JSON object to the ServiceNow CMDBGroupAPI, which sets the manual CI list of the group.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Example (Request URL)

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/cmdb_group
```

Example (Body)

```
[
  {
    "items": [
      {
        "name": "test",
        "description": "",
        "group": "",
        "manualCIList":
          "d83dac0adb4dab00dc44f00fbf961919,2e6b7046db8dab00dc44f00fbf961929,7fb39667db
          a
          12380dc44f00fbf961901,77b39667dba12380dc44f00fbf961917,7bb39667dba12380dc44f0
          0
          fbf96191c",
        "region": "Cisco",
      }
    ]
  }
]
```

```
        "id": "1"
      }
    ]
  }
]
```

Example (Response)

```
{
  "result": [
    {
      "idList": [
      ],
      "partialCIListDueToACLFlag": false,
      "nextBatchStart": 0,
      "result": true
    },
    {
      "idList": [
      ],
      "partialCIListDueToACLFlag": false,
      "nextBatchStart": 0,
      "result": true
    }
  ]
}
```

Companies

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/companies`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/companies`

This operation supports Domain Separation enabled or not enabled. This operation pulls all the fields for from the company table that are not NULL values. The return is ordered by **sys_id**, so the results display in the same order every time. The results are filtered by the **SL1 Monitored** and **region** values. The region must be supplied by the requester, and it will only return region-specific companies.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic
domainSep	false
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status

Code	Value
200	OK
400	Query parameter \region\ is not defined and is required.

Fixed Internal Query

Domain:

```
'x_sclo_scilogic_region=' + region + '^x_sclo_scilogic_monitored=true^sys_
domain!=global'
```

Non-Domain:

```
'x_sclo_scilogic_monitored=true^x_sclo_scilogic_idISNOTEMPTY^x_sclo_scilogic_region''
+ region
```

Example

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/
companies?region=ScienceLogic&sysparm_offset=0&sysparm_limit=100
```

Example (Response)

```
{
  "results": [
    {
      "country": "USA",
      "notes": "What's on your digital horizon?",
      "city": "San Jose",
      "sys_updated_on": "2018-11-30 16:03:45",
      "sys_class_name": "core_company",
      "sys_id": "1ac84f95dbce2700dc44f00fbf9619c8",
      "sys_updated_by": "is4user1",
      "market_cap": "0",
      "street": "170 West Tasman Dr.",
      "sys_created_on": "2018-11-27 16:32:33",
      "state": "CA",
      "sys_created_by": "admin",
      "zip": "95134",
      "profits": "0",
      "revenue_per_year": "0",
      "sys_mod_count": "4",
      "x_sclo_scilogic_id": "1",
      "x_sclo_scilogic_monitored": "true",
      "phone": "18005532447",
      "name": "Cisco Systems, Inc.",
      "x_sclo_scilogic_region": "Cisco"
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 1,
  "return_count": 1,
  "total_count": 1
}
```

Configuration Items

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/configuration_item`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/configuration_item`

This operation pulls all the fields for any configuration item or class that are not Null values. The return is ordered by **sys_id**, so the results display in the same order every time. The results are filtered by the **SL1 monitored** and **SL1 ID** field on the ServiceNow side. This operation requires the region to be supplied by the requester and will only return region-supplied configuration items. Two classes have been filtered out: the network adapter (**cmdb_ci_network_adapter**) and file system (**cmdb_ci_file_system**). All devices require specific fields that must be populated to be included in the query. Two specific classes, the network adapter and file systems, have been separated out from the main configuration item pull. All requests work the same way. On the response, the operation returns all non-empty fields associated with each configuration item.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status	
Code	Value
200	OK
400	Query Parameter \region\ is not defined and is required.

Fixed Internal Query

```
'x_sclo_scilogic_monitored=true^x_sclo_scilogic_idISNOTEMPTY^x_sclo_scilogic_region='
+ region + 'sys_class_name!=cmdb_ci_network_adapter^ORsys_class_name!=cmdb_file_
system'
```

Example

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/
configuration_Items?region=ScienceLogic&sysparm_offset=5&sysparm_limit=11
```

Example (Response)

```
{
  "results": [
    {
      "operational_status": "1",
      "sys_updated_on": "2018-12-13 12:16:17",
      "discovery_source": "Other Automated",
      "first_discovered": "2018-10-10 21:06:23",
      "sys_updated_by": "admin",
      "sys_domain": "global",
      "sys_created_by": "is4user1",
      "sys_domain_path": "/",
      "install_status": "1",
      "name": "Test Class 1",
      "subcategory": "Computer",
      "last_discovered": "2018-12-13 12:16:17",
      "sys_class_name": "cmdb_ci_linux_server",
      "sys_id": "2e6b7046db8dab00dc44f00fbf961929",
      "sys_class_path": "/!!!/2/!(/!!!/0",
      "mac_address": "77:55:33:4C:2A:A3",
      "sys_mod_count": "1",
      "x_sclo_scilogic_id": "200",
      "serial_number": "HGFE6789",
      "model_id": "143d608edb4dab00dc44f00fbf9619f6",
      "ip_address": "10.10.10.0",
      "cost_cc": "USD",
      "x_sclo_scilogic_monitored": "true",
      "asset": "666b7046db8dab00dc44f00fbf96192a",
      "category": "Hardware",
      "x_sclo_scilogic_region": "ScienceLogic"
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 1,
  "return_count": 1,
  "total_count": 16
}
```

Device Identification Engine

HTTP Method

POST

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/IdentificationEngine`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/IdentificationEngine`

This operation handles all creates and updates to the CMDB. This operation incorporates Identification Engine and uses the Identification and Reconciliation framework to properly import devices into the CMDB as a configurable discovery source. SL1 uses the classification GET to populate the JSON object.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
test	true

Example (Request URL)

`https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/IdentificationEngine?test=true`

Example (Body)

```
[
  {
    "items": [
      {
        "className": "cmdb_ci_linux_server",
        "values": {
```

```

        "name": "Postman Test Server 1",
        "serial_number": "9876EFGH",
        "mac_address": "BF:D4:D6:6E:56:F1",
        "ip_address": "10.10.10.4",
        "ram": "16000",
        "x_sclo_scilogic_region": "ScienceLogic",
        "x_sclo_scilogic_id": "1"
    }
}
],
{
    "items": [
        {
            "className": "cmdb_ci_linux_server",
            "values": {
                "name": "Postman Test Server 2",
                "serial_number": "HGFE6789",
                "mac_address": "87:54:3C:8C:2A:A3",
                "ip_address": "10.10.10.5",
                "ram": "16000",
                "x_sclo_scilogic_region": "ScienceLogic",
                "x_sclo_scilogic_id": "2"
            }
        }
    ]
}
]

```

Example Business Service (Body)

```

[
  {
    "items": [
      {
        "className": "cmdb_ci_service",
        "values": {
          "name": "Integration Service",
          "busines_criticality": "1 - most critical",
          "used_for": "Production",
          "operational_status": "1",
          "service_classification": "Technical Service",
          "comments": "Postman",
          "x_sclo_scilogic_region": "ScienceLogic",
          "x_sclo_scilogic_id": "1570"
        }
      },
      {
        "className": "cmdb_ci_linux_server",
        "values": {
          "name": "Postman Test Server",
          "serial_number": "7MDvqrSNyd",
          "manufacturer": "ScienceLogic, Inc.",
          "model_id": "",
          "mac_address": "EE:D6:0B:79:32:C7",
          "ip_address": "10.10.10.224",
          "ram": "16000",
          "x_sclo_scilogic_region": "ScienceLogic",
          "x_sclo_scilogic_id": "10"
        }
      }
    ]
  }
]

```

```

    }
  ],
  "relations": [
    {
      "type": "Depends on::Used by",
      "parent": 0,
      "child": 1
    }
  ]
}
]

```

Example (Response)

```

{
  "result": [
    {
      "items": [
        {
          "className": "cmdb_ci_linux_server",
          "operation": "NO_CHANGE",
          "sysId": "7fb39667dba12380dc44f00fbf961936",
          "identifierEntrySysId": "fb27f69cc3000200d8d4bea192d3ae67",
          "identificationAttempts": [
            {
              "identifierName": "Hardware Rule",
              "attemptResult": "SKIPPED",
              "attributes": [
                "serial_number",
                "serial_number_type"
              ],
              "searchOnTable": "cmdb_serial_number"
            },
            {
              "identifierName": "Hardware Rule",
              "attemptResult": "MATCHED",
              "attributes": [
                "serial_number"
              ],
              "searchOnTable": "cmdb_ci_hardware"
            }
          ]
        }
      ]
    }
  ],
  "relations": [
    ]
  ]
}

```

File Systems

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/file_systems`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/file_systems`

This operation pulls all the fields from the File System table. The return is ordered by **sys_id**, so the results display in the same order every time. The results are filtered by the **SL1 monitored** and **SL1 ID** field on the ServiceNow side. This operation requires the region to be supplied by the requester, it returns only region-supplied configuration items.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status	
Code	Value

200	OK
400	Query Parameter \region\ is not defined and is required.

Fixed Internal Query

```
`x_sclo_scilogic_monitored=true^x_sclo_scilogic_idISNOTEMPTY^x_sclo_scilogic_region='
+ region
```

Example

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/file_
systems?region=ScienceLogic&sysparm_offset=0&sysparm_limit=100
```

Example (Response)

```
{
  "results": [
    {
      "operational_status": "1",
      "sys_updated_on": "2018-11-12 21:59:52",
      "media_type": "fixed",
      "sys_created_by": "admin",
      "sys_domain_path": "/",
      "sys_class_name": "cmdb_ci_file_system",
      "computer": "d83dac0adb4dab00dc44f00fbf961919",
      "x_sclo_scilogic_monitored": "true",
      "x_sclo_scilogic_region": "ScienceLogic",
      "sys_updated_by": "admin",
      "sys_created_on": "2018-11-12 21:59:06",
      "sys_domain": "global",
      "install_status": "1",
      "name": "/root",
      "subcategory": "File Share",
      "sys_id": "afd30ba0dbf5a380dc44f00fbf961951",
      "file_system": "ntfs",
      "sys_class_path": "/!!!/!K/!!",
      "mount_point": "/root",
      "sys_mod_count": "3",
      "x_sclo_scilogic_id": "31",
      "label": "/root",
      "cost_cc": "USD",
      "category": "Resource"
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 100,
  "return_count": 1,
  "total_count": 1
}
```

Import Set

HTTP Method

POST

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/import_set`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/import_set`

This operation handles the custom intake of import sets before it reaches the transform map staging table, such as `x_sclo_scilogic_import_installed_software`. This operations is currently only used for importing installed software (`x_sclo_scilogic_import_installed_software`).

Headers	
Key	Value
Accept	application/json
Content-Type	application/json

Parameters	
Key	Value
record_type (required)	x_sclo_scilogic_import_installed_software

Example (Request URL)

`https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/import_set`

Example (Body)

```
[
  {
    "records": [
      {
        "name": "acl-2.2.51-12.e17",
        "software": "671bafd8dba13700dc44f00fbf961953",
        "cmdb_ci": [
          "ff01a81edb1df300dc44f00fbf961947",
          "4011a81edb1df300dc44f00fbf961958",
          "f301a81edb1df300dc44f00fbf96193d",
          "7b01a81edb1df300dc44f00fbf961942",
          "c411a81edb1df300dc44f00fbf96195d",
```

```
    "7701a81edb1df300dc44f00fbf961922",
    "7b01681edb1df300dc44f00fbf9619e7",
    "fb01a81edb1df300dc44f00fbf961927"
  ],
  "active": true
}
]
]
```

Incidents

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/incidents`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/incidents`

This operation pulls all records from the incident table that are created by a specific **user_id** and its related events. The results are ordered by the **sys_id** of the incident, so the results display in the same order every time. This operation is also based on the incident being in an active state. This operation returns a pre-set of data and does not return everything on the Incident and Event (x_sclo_scilogic_event) tables.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
user_id (required)	is4user1
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status	
Code	Value

200	OK
400	Query Parameter '\user_id\' is not defined and is required.

Fixed Internal Query

```
'sys_created_by=' + user_id + 'active=true'
```

Example

```
https://<your Instance>.service-now.com/api/x_sclo_  
scilogic/v1/sciencelogic/incidents?user_id=is4user1&sysparm_offset=0&sysparm_limit=100
```

Example (Response)

```
{  
  "results": [  
    {  
      "sys_id": "0141807bdbb16300dc44f00fbf9619fc",  
      "number": "INC0010135",  
      "state": "2",  
      "state_label": "In Progress",  
      "events": [  
        {  
          "event_id": "16908",  
          "device": {  
            "sys_id": {  
            }  
          }  
        },  
        {  
          "event_id": "16874",  
          "device": {  
            "sys_id": {  
            }  
          }  
        },  
        {  
          "event_id": "16865",  
          "device": {  
            "sys_id": {  
            }  
          }  
        }  
      ]  
    },  
    "sysparm_offset": 0,  
    "sysparm_limit": 1,  
    "return_count": 1,  
    "total_count": 44  
  ]  
}
```

Installed Software

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/installed_software`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/installed_software`

This operation pulls all the fields from the software (cmdb_ci_spkg) table. The return is ordered by **sys_id**, so the results display in the same order every time. The results are filtered by the **SL1 monitored** field on the ServiceNow side. This operation requires the **region** to filter the installed software on devices.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region	ScienceLogic
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status	
Code	Value
200	OK

400

Query parameter `\'region\'` is not defined and are required.

Fixed Internal Query

`\'x_sclo_scilogic_monitored=true\'`

Example (Request URL)

`https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/installed_software?sysparm_offset=0&sysparm_limit=100®ion=ScienceLogic`

Example (Response)

```
{
  "results": [
    {
      "operational_status": "1",
      "operational_status_label": "Operational",
      "sys_updated_on": "2019-05-01 06:00:09",
      "install_count": "2",
      "sys_updated_by": "system",
      "sys_created_on": "2019-03-29 19:42:58",
      "sys_domain": "global",
      "sys_created_by": "admin",
      "sys_domain_path": "/",
      "install_status": "1",
      "install_status_label": "Installed",
      "name": "Test_31",
      "subcategory": "Package",
      "sys_class_name": "cmdb_ci_spkg",
      "sys_class_name_label": "Software",
      "sys_id": "1e9608fcdb2cb740dc44f00fbf961949",
      "sys_class_path": "/!/#$",
      "key": "Test_31::_NULL",
      "license_available": "-2",
      "sys_mod_count": "1",
      "x_sclo_scilogic_id": "31",
      "model_id": "2c146728dbe8b740dc44f00fbf9619c6",
      "model_id_label": "Unknown",
      "cost_cc": "USD",
      "cost_cc_label": "USD",
      "x_sclo_scilogic_monitored": "true",
      "package_name": "Test_31",
      "category": "Software",
      "x_sclo_scilogic_region": "AutoGenerateClass",
      "installed_on": [
        {
          "sys_id": "5a271407dbfe6300dc44f00fbf96190f",
          "id": "10",
          "region": "ScienceLogic",
          "monitored": "true"
        },
        {
          "sys_id": "5a271407dbfe6300dc44f00fbf96190f",
          "id": "10",

```

```
        "region": "ScienceLogic",
        "monitored": "true"
    }
]
}
],
"sysparm_offset": 0,
"sysparm_limit": 100,
"return_count": 4,
"total_count": 4
}
```

Manufacturer

HTTP Method

POST

Pagination

Enabled

Resource Path

```
/api/x_sclo_scilogic/v1/sciencelogic/manufacture
```

Default Resource Path

```
/api/x_sclo_scilogic/sciencelogic/manufactures
```

This operation does not populate any data into ServiceNow. Instead, this operation takes an array of manufacturer names and attempts to line them up with manufacturers already in ServiceNow. Then the operation returns the sys_id of manufacturers it was able find based on matching name. If the Normalization Data Services Client is active on the target instance, this operation uses those tables to find a matching company record; otherwise the operation will match on whether name and manufacturer is true on the core_company table.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic

Example (Request URL)

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/manufacture
```

Example (Body)

```
{
  "manufactures": [
    "Cisco Systems Inc",
    "Cisco Systems, Incorporated",
```

```
"CiscoSystems",
"American Power Conversion Inc.",
"APC Corp",
"Apc",
"IBM",
"IBM CORP",
"International Business Machines",
"Juniper Systems",
"Juniper Networks,Inc",
"Juniper Solutions"
]
}
```

Example (Response)

```
{
  "result": {
    "Cisco Systems Inc": "",
    "Cisco Systems, Incorporated": "",
    "CiscoSystems": "",
    "American Power Conversion Inc.": "",
    "APC Corp": "",
    "Apc": "",
    "IBM": "",
    "IBM CORP": "",
    "International Business Machines": "",
    "Juniper Systems": "",
    "Juniper Networks,Inc": "",
    "Juniper Solutions": ""
  }
}
```

Model

HTTP Method

POST

Resource Path

```
/api/x_sclo_scilogic/v1/sciencelogic/model
```

Default Resource Path

```
/api/x_sclo_scilogic/sciencelogic/models
```

This operation does not populate any data into ServiceNow. Instead, this operation takes an array of model names and attempts to line them up with models already in ServiceNow and returns the sys_id of models it was able to find based on matching name.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic

Example (Request URL)

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/models
```

Example (Body)

```
{
  "models": [
    "4331 ISR",
    "7206VXR",
    "7609S",
    "AS5300",
    "ASR5000",
    "Catalyst 3560G-24TS",
    "Catalyst 4948",
    "Catalyst 6509-CatOS",
    "BIG-IP Viprion B4300",
    "F5 BIG-IP DNS",
```

```
"BIG-IP Wide IP Container",  
"BIG-IP Data Center Container"  
]  
}
```

Example (Response)

```
{  
  "result": {  
    "4331 ISR": "",  
    "7206VXR": "",  
    "7609S": "",  
    "AS5300": "",  
    "ASR5000": "",  
    "Catalyst 3560G-24TS": "",  
    "Catalyst 4948": "",  
    "Catalyst 6509-CatOS": "",  
    "BIG-IP Viprion B4300": "",  
    "F5 BIG-IP DNS": "",  
    "BIG-IP Wide IP Container": "",  
    "BIG-IP Data Center Container": ""  
  }  
}
```

Network Adapters

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/network_adapters`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/network_adapters`

This operation pulls all the fields from the network adapter table. The return is ordered by **sys_id**, so the results display in the same order every time. The results are filtered by the **SL1 monitored** and **SL1 ID** field on the ServiceNow side. This operation requires the region to be supplied by the requester, and it only returns region-supplied configuration items.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status	
Code	Value

200	OK
400	Query Parameter '\region\' is not defined and is required.

Fixed Internal Query

```
`x_sclo_scilogic_monitored=true^x_sclo_scilogic_idISNOTEMPTY^x_sclo_scilogic_region='
+ region
```

Example (Request URL)

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/network_
adapters?region=ScienceLogic&sysparm_offset=0&sysparm_limit=100
```

Example (Response)

```
{
  "results": [
    {
      "operational_status": "1",
      "sys_updated_on": "2018-11-12 21:29:23",
      "sys_updated_by": "admin",
      "sys_created_on": "2018-11-12 21:27:48",
      "sys_domain": "global",
      "sys_created_by": "admin",
      "cldb_ci": "d83dac0adb4dab00dc44f00fbf961919",
      "sys_domain_path": "/",
      "install_status": "1",
      "name": "eth0",
      "subcategory": "Network",
      "sys_class_name": "cldb_ci_network_adapter",
      "sys_id": "33ac36acdbb5a380dc44f00fbf961963",
      "netmask": "255.255.255.0",
      "sys_class_path": "/!//!8",
      "mac_address": "BF:D4:D6:6E:56:F1",
      "sys_mod_count": "3",
      "x_sclo_scilogic_id": "20",
      "ip_address": "10.10.10.4",
      "cost_cc": "USD",
      "x_sclo_scilogic_monitored": "true",
      "category": "Hardware",
      "x_sclo_scilogic_region": "ScienceLogic"
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 1,
  "return_count": 1,
  "total_count": 5
}
```

Service Requests

HTTP Method

GET

Pagination

Enabled

Resource Path

`/api/x_sclo_scilogic/v1/sciencelogic/service_request`

Default Resource Path

`/api/x_sclo_scilogic/sciencelogic/service_request`

This operation pulls all service requests that are tied to specific catalog item. Based on the request type it returns a formatted JSON object. It pulls all the required information for an SL1 Discovery session and creating a virtual device in SL1. Both requests require different information and are formatted accordingly.

The basic catalog item Device Discovery is set up as information collection to support the process within SL1. The Service Catalog has been simplified to its most basic form. The workflow moves the request into the correct state to be picked up by the GET request and then waits for its return before completing the workflow.

Headers	
Key	Value
Content-Type	application/json
Accept	application/json

Parameters	
Key	Value
region (required)	ScienceLogic
state	2
sysparm_offset	0
sysparm_limit	glide.json.export.limit, glide.ui.export.limit, glide.ui.export.war.threshold

HTTP Status	
Code	Value
200	OK
400	Query Parameter \region\ is not defined and is required.

Fixed Internal Query

State:

```
`request_item.active=true^request_item.cat_item=' + catalog + `^sc_item_option.item_option_new.name=Region^sc_item_option.value=' + region
```

Non-State:

```
`request_item.active=true^request_item.cat_item=' + catalog + `^sc_item_option.item_option_new.name=Region^sc_item_option.value=' + region + `^request_item.state=' + state
```

Example

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v1/sciencelogic/ service_request?region=Cisco
```

Example (Response)

```
{
  "results": [
    {
      "number": "RITM0010018",
      "sysid": "00365de2db1a2340dc44f00fbf961941",
      "state": "2",
      "request_type": "Discover Device",
      "region": "Cisco",
      "log_all": "false",
      "ip_hostname_list": "167.132.14.15",
      "credentials": [
        {
          "Category": "Linux",
          "ID": "1"
        }
      ]
    },
    {
      "discover_non_snmp": "false",
      "model_devices": "true",
      "dhcp": "false",
      "device_model_cache_ttl_h": "2",
      "collection_server": "1",
      "organization": "1",
      "add_devices_to_device_groups": [
        "test"
      ]
    },
    {
      "device_template": "1",
      "initial_scan_level": "System Default (Recommended)",
      "scan_throttle": "System Default (Recommended)",
      "scan_ports": "21,22,23,25,80",
    }
  ]
}
```

```

    "port_scan_all": "System Default (Recommended)",
    "port_scan_timeout": "System Default (Recommended)",
    "interface_inventory_timeout": "600000",
    "maximum_allowed_interfaces": "10000",
    "bypass_interface_inventory": "false"
  },
  {
    "number": "RITM0010016",
    "sysid": "194447e8db162f00dc44f00fbf96195b",
    "state": "2",
    "request_type": "Discover Device",
    "region": "Cisco",
    "log_all": "false",
    "ip_hostname_list": "192.168.1.1",
    "credentials": [
      {
        "Category": "Linux",
        "ID": "1"
      }
    ]
  },
  {
    "discover_non_snmp": "false",
    "model_devices": "false",
    "dhcp": "false",
    "device_model_cache_ttl_h": "2",
    "collection_server": "1",
    "organization": "1",
    "add_devices_to_device_groups": [
      {
        "device_template": "1",
        "initial_scan_level": "System Default (Recommended)",
        "scan_throttle": "System Default (Recommended)",
        "scan_ports": "21,22,23,25,80",
        "port_scan_all": "System Default (Recommended)",
        "port_scan_timeout": "System Default (Recommended)",
        "interface_inventory_timeout": "600000",
        "maximum_allowed_interfaces": "10000",
        "bypass_interface_inventory": "false"
      }
    ]
  },
  {
    "number": "RITM0010014",
    "sysid": "250dae2cdbc22f00dc44f00fbf961954",
    "state": "2",
    "request_type": "create_virtual_device",
    "region": "Cisco",
    "collection_server": "1",
    "virtual_device_class": "1"
  }
]
}

```

Appendix

D

ServiceNow Registered Events

Overview

This appendix describes the commands and data you can use to generate registered events in ServiceNow that are queued to ServiceNow Event Management. These events can trigger actions in the Integration Service, such as specifying one or more CIs for monitoring, or putting a CI into maintenance.

These events use the `gs.eventQueue` command, using the following format:

```
eventQueue(String name, Object instance, String parm1, String parm2)
```

You can use examples found in the following ServiceNow update sets to help you customize the `gs.eventQueue` command to specify which ServiceNow events can trigger Integration Service actions:

- ScienceLogic ServiceNow Integration (Catalog UI)
- ScienceLogic ServiceNow Integration (Maintenance Mode)

You will need to install these update sets in ServiceNow.

NOTE: This appendix is recommended for advanced ServiceNow administrators.

This appendix includes the following topics:

Catalog Item Events	213
Maintenance Mode Events	215

Catalog Item Events

The following events are available through the "ScienceLogic ServiceNow Integration (Catalog UI)" update set in ServiceNow.

x_sclo_scilogic.device_monitoring

This event takes the selected Configuration Items in ServiceNow, files a catalog request using the template selected by the user, and submits the catalog request.

Trigger

Custom requirement supplied by ScienceLogic implementation or the Customer directly.

Command

```
gs.eventQueue('x_sclo_scilogic.device_monitoring', region, ip_list.toString(),  
region.getUniqueValue() + "," + region.x_sclo_scilogic_region + "," + silo_template);
```

Event Fields

Name	Type	Field	Description
Name	String	x_sclo_scilogic.device_monitoring	Unique name of the event.
glideRecord	Object	region	The table to which the event applies.
parm1	String	ip_list.toString()	Parm1: The IP, or a comma-separated list of IP addresses, that is pulled from the ip_address field on the cmdb_ci table.
parm2	String	getCompany.getUniqueValue(), silo_template	Parm2: List of three requirements that the sys_id of the company associated with the Configuration Item and the catalog template selected through the user interface action.

Example

The UI action / UI page is available through the ScienceLogic ServiceNow Integration (Catalog UI Action) update set.

x_sclo_scilogic.remove_monitoring

This action takes the selected Configuration Item or Items and submits a request through the ServiceNow service catalog for each Configuration Item.

Trigger

Custom requirement supplied by ScienceLogic implementation or the Customer directly.

Command

```
gs.eventQueue('x_sclo_scilogic.remove_monitoring',current, current.getUniqueValue(),  
current.company);
```

Event Fields

Name	Type	Field	Description
Name	String	x_sclo_scilogic.remove_monitoring	Unique name of the event.
glideRecord	Object	current	The table to which the event applies.
parm1	String	current.getUniqueValue()	Parm1: The sys_id of the Configuration Item that needs to be removed
parm2	String	current.company);	Parm2: The sys_id of the company that is associated with the Configuration Item.

Example

The UI action / UI page is available through the ScienceLogic ServiceNow Integration (Catalog UI Action) update set.

Maintenance Mode Events

The following event is available through the "ScienceLogic ServiceNow Integration (Maintenance Mode)" update set in ServiceNow.

x_sclo_scilogic.device_maintenance

This event submits a list of devices to be put in to Maintenance via Mid Server.

Trigger

Custom requirement supplied by ScienceLogic implementation or the Customer directly.

Command

```
gs.eventQueue('x_sclo_scilogic.device_maintenance',current, action, affected_ci);
```

Event Fields

<i>Name</i>	<i>Type</i>	<i>Field</i>	<i>Description</i>
Name	String	x_sclo_scilogic.device_maintenance	Unique name of the event.
glideRecord	Object	current	The table to which the event applies.
parm1	String	action	Parm1: An array that includes action to be performed (enable_maintenance or disable_maintenance) and the sys_id of the task. Task is not required the action is.
parm2	String	affected_ci	Parm2: An array of device sys_ids that need to be enable or disabled maintenance mode.

Example

Business rule (ScienceLogic ServiceNow Integration (Maintenance Mode Business rule)).

© 2003 - 2020, ScienceLogic, Inc.

All rights reserved.

LIMITATION OF LIABILITY AND GENERAL DISCLAIMER

ALL INFORMATION AVAILABLE IN THIS GUIDE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. SCIENCELOGIC™ AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.

Although ScienceLogic™ has attempted to provide accurate information on this Site, information on this Site may contain inadvertent technical inaccuracies or typographical errors, and ScienceLogic™ assumes no responsibility for the accuracy of the information. Information may be changed or updated without notice. ScienceLogic™ may also make improvements and / or changes in the products or services described in this Site at any time without notice.

Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: legal@sciencelogic.com



ScienceLogic

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010