



---

# ServiceNow Change Management SyncPack

Version 4.1.0

---

# Table of Contents

<b>Introduction to the ServiceNow Change Management SyncPack</b> .....	<b>4</b>
What Can I Do with this SyncPack? .....	4
Contents of the SyncPack .....	5
PowerFlow Applications .....	5
PowerFlow Applications (Internal) .....	6
<b>Installing the Change Management SyncPack</b> .....	<b>7</b>
Architecture Overview for ServiceNow SyncPacks .....	8
SL1 and ServiceNow Terminology .....	9
Dependency Map for ServiceNow SyncPacks .....	9
Prerequisites for ServiceNow SyncPacks .....	9
Downloading the SyncPack .....	10
Importing the SyncPack .....	11
Installing the SyncPack .....	11
<b>Configuring Applications for the Change Management SyncPack</b> .....	<b>13</b>
Creating and Aligning a Configuration Object .....	14
Creating a Configuration Object .....	14
Aligning a Configuration Object .....	17
Creating an OAuth2 Credential Record in ServiceNow .....	18
Syncing Device Maintenance from ServiceNow to SL1 .....	19
Viewing SL1 Events for Change Requests .....	19
Viewing Change Events for SL1 Business Services .....	20
Scheduling Device Maintenance (No MID Server) .....	21
Using a MID Server to Place a Device Directly in Maintenance or Schedule a Maintenance Window .....	26
Configuring and Running the "Sync Change Management Requirements" Application in PowerFlow .....	26
Creating a ServiceNow Credential and Credential Affinity .....	27
Configuring ServiceNow to use an API Key for PowerFlow Authentication .....	28
ServiceNow Registered Events .....	29
Change Request Events .....	29
Direct to Device: x_sclo_scilogic.device_maintenance .....	29
Command .....	29
Event Fields .....	30

Scheduled Device Maintenance: x_sclo_scilogic.device_maintenance_skd .....	30
Command .....	30
Event Fields .....	30
Change Request Events Example .....	31
<b>Troubleshooting the Change Management SyncPack .....</b>	<b>33</b>
Initial Troubleshooting Steps .....	34
SL1 PowerFlow .....	34
ServiceNow .....	34
Resources for Troubleshooting .....	34
Useful PowerFlow Ports .....	34
Helpful Docker Commands .....	35
Viewing Container Versions and Status .....	35
Restarting a Service .....	35
Stopping all PowerFlow Services .....	35
Restarting Docker .....	35
Diagnosis Tools .....	36
Retrieving Additional Debug Information (Debug Mode) .....	36
Allowing Cross-Scoped Access in ServiceNow .....	38
<b>ServiceNow Change Management API Endpoints .....</b>	<b>41</b>
Change Requests .....	42
HTTP Method .....	42
Resource Path .....	42
Pagination .....	42
Overview .....	42
Example (Request URL) .....	43
Examples (Response) .....	43

---

# Chapter

# 1

## Introduction to the 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.

You do not need the ServiceNow Events Module to use this SyncPack. However, you will need to install and run the "ServiceNow CMDB" SyncPack on the same PowerFlow system as this SyncPack.

This chapter covers the following topics:

<a href="#">What Can I Do with this SyncPack?</a> .....	4
<a href="#">Contents of the SyncPack</a> .....	5

---

### What Can I Do with this SyncPack?

The "ServiceNow Change Management" SyncPack 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.

You can configure PowerFlow to generate an SL1 event when a maintenance-related change occurs. The SL1 event contains data about the change request from ServiceNow, including a hyperlink to the ServiceNow change record. The event is aligned with a device in SL1, and if the device is part of a Business Service in SL1 version 11.1.0 or later, the event displays on the **[Change Events]** tab for that Business Service. This sync of proposed ServiceNow change requests to SL1 events requires the ServiceNow MID Server.

**NOTE:** You do not need the ServiceNow Events Module to use this SyncPack. However, you will need to install and run the "ServiceNow CMDB" SyncPack on the same PowerFlow system as this SyncPack.

This SyncPack includes the following applications:

- **Create or Update Maintenance Schedule from ServiceNow Trigger.** Handles scheduling and canceling maintenance schedules in SL1 using a trigger from a ServiceNow Management, Instrumentation, and Discovery (MID) Server. When triggered, the application gathers data about the change request in ServiceNow and generates an SL1 Event with a severity of Notice. For more information, see [Syncing Device Maintenance from ServiceNow to SL1](#).
- **Sync Change Management Requirements.** Sends configuration data from PowerFlow to ServiceNow to use with change management and other processes. For more information, see [Configuring and Running the "Sync Change Management Requirements" Application in PowerFlow](#).
- **Sync Maintenance Schedules from ServiceNow to SL1.** 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. If you are scheduling this application, ScienceLogic recommends that you schedule it to run every hour or so, depending on your environment. This process does *not* require the use of a ServiceNow MID Server. For more information, see [Scheduling Device Maintenance \(No MID Server\)](#).
- **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 [Using a MID Server to Place a Device Directly in Maintenance or Schedule a Maintenance Window](#).

---

## Contents of the SyncPack

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

### PowerFlow Applications

The following PowerFlow applications are included with the "ServiceNow Change Management" SyncPack:

- **Create or Update Maintenance Schedule from ServiceNow Trigger.** Handles scheduling and canceling maintenance schedules in SL1 using a trigger from a ServiceNow Management, Instrumentation, and Discovery (MID) Server. When triggered, the application gathers data about the change request in ServiceNow and generates an SL1 Event with a severity of Notice. For more information, see [Syncing Device Maintenance from ServiceNow to SL1](#).
- **Sync Change Management Requirements.** Sends configuration data from PowerFlow to ServiceNow to use with change management and other processes. For more information, see [Configuring and Running the "Sync Change Management Requirements" Application in PowerFlow](#).
- **Sync Maintenance Schedules from ServiceNow to SL1.** 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. If you are scheduling this application, ScienceLogic recommends that you schedule it to run every hour or so, depending on your environment. This process does *not* require the use of a ServiceNow MID Server. For more information, see [Scheduling Device Maintenance \(No MID Server\)](#).

- **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 [Using a MID Server to Place a Device Directly in Maintenance or Schedule a Maintenance Window](#).

## PowerFlow Applications (Internal)

To view the internal PowerFlow applications, click the Filter icon (☰) on the **Applications** page and select *Show Hidden Applications*. Internal applications are hidden by default. The following 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.
- **Process Create Schedule Requests.** Schedules and posts maintenance tasks in ServiceNow.
- **Remove Maintenance.** Removes a scheduled maintenance in SL1.
- **Schedule Maintenance.** Creates a scheduled maintenance in SL1.

## Installing the Change Management SyncPack

---

### Overview

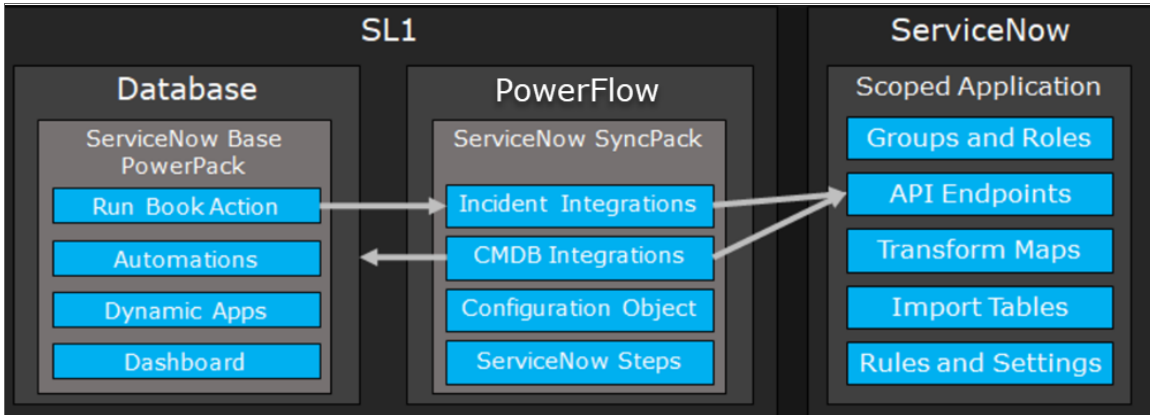
This chapter describes how to install the "ServiceNow Change Management" SyncPack.

This chapter covers the following topics:

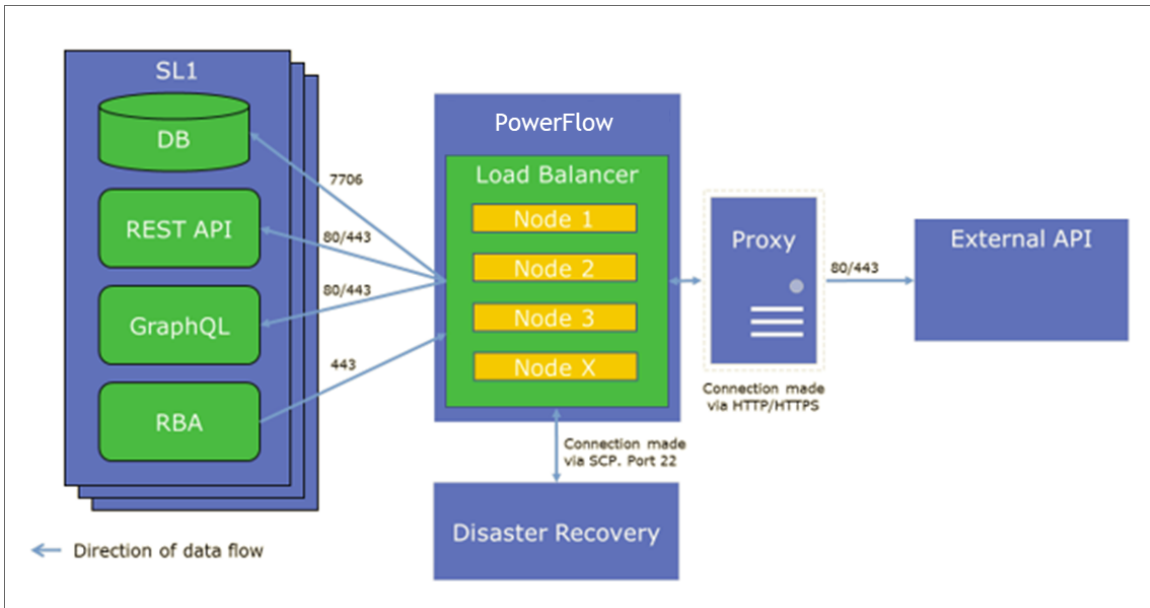
<i>Architecture Overview for ServiceNow SyncPacks</i> .....	8
<i>SL1 and ServiceNow Terminology</i> .....	9
<i>Dependency Map for ServiceNow SyncPacks</i> .....	9
<i>Prerequisites for ServiceNow SyncPacks</i> .....	9
<i>Downloading the SyncPack</i> .....	10
<i>Importing the SyncPack</i> .....	11
<i>Installing the SyncPack</i> .....	11

# Architecture Overview for ServiceNow SyncPacks

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



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





---

## SL1 and ServiceNow Terminology

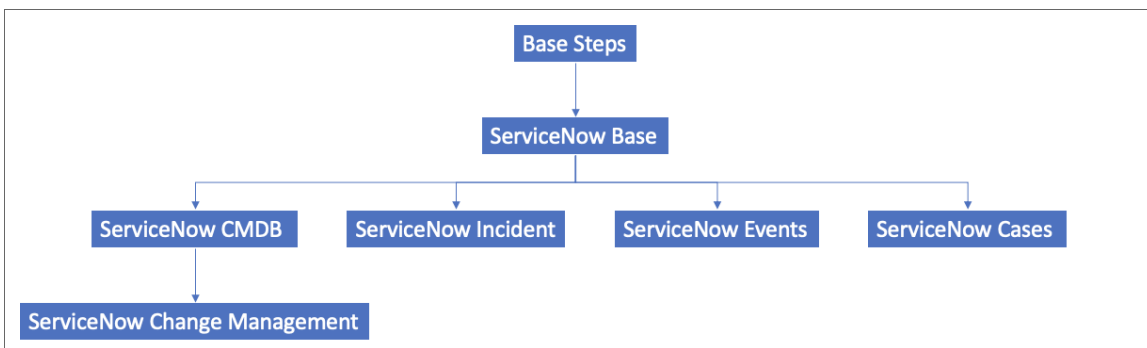
The following table lists the different names for the shared elements in SL1 and ServiceNow:

SL1	ServiceNow
Asset, Custom Attribute	Asset (ITAM)
Device	CI (Configuration Item)
Discovery Session	Service Request, Catalog Request
Event	Incident, Event, or Case (depending on the SyncPack you are using)
Alert	Event
Organization	Company, Domain
Schedule, Maintenance Schedule	Change Request, Change Schedule
Topology, Relationships, Dynamic Component Mapping and Relationships (DCM+R)	Dependency View, Affected CIs

---

## Dependency Map for ServiceNow SyncPacks

The following graphic describes which SyncPacks depend on other SyncPacks:



**TIP:** For more information about the "Base Steps" SyncPack, see the *SL1 PowerFlow 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 release notes for that SyncPack.

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

- ScienceLogic administrator access to the Administration Portal
- ServiceNow administrator access

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.

**WARNING:** ScienceLogic does not support any deployment that attempts to sync one SL1 instance to multiple ServiceNow instances. A deployment of this type will be incredibly fragile and would require the customer to strictly control their environments. This is not something that can be controlled programmatically. Escalations related to this type of deployment are not supported.

The following table lists the port access required by PowerFlow and this SyncPack:

Source IP	PowerFlow Destination	PowerFlow Source Port	Destination Port	Requirement
PowerFlow	SL1 API	Any	TCP 443	SL1 API Access
PowerFlow	ServiceNow API	Any	TCP 443	ServiceNow API Access
PowerFlow	SL1 Database	Any	TCP 7706	SL1 Database Access

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

---

## Downloading the SyncPack

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

**NOTE:** If you are installing or upgrading to the latest version of this SyncPack in an offline deployment, see [Installing or Upgrading in an Offline Environment](#) to ensure you install any external dependencies.

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 and select *PowerPacks & SyncPacks*.
3. In the **Search** field, search for the SyncPack and select it from the search results. The **Release Version** page appears.

4. On the **[Files]** tab, click the down arrow next to the SyncPack version that you want to install, and select *Show File Details*. The **Release File Details** page appears.
5. Click the **[Download File]** button to download the SyncPack.

After you download the SyncPack, you can import it to your PowerFlow system using the PowerFlow user interface.

---

## Importing the SyncPack

**NOTE:** You must import and install the "ServiceNow Base" SyncPack before uploading and installing this SyncPack.

To import a SyncPack in the PowerFlow user interface:

1. On the **SyncPacks** page (☺) of the PowerFlow user interface, click **[Import SyncPack]**. The **Import SyncPack** page appears.
2. Click **[Browse]** and select the **.whl** file for the SyncPack you want to install. You can also drag and drop a **.whl** file to the **Import SyncPack** page.
3. Click **[Import]**. PowerFlow registers and uploads the SyncPack. The SyncPack is added to the **SyncPacks** page.
4. You will need to activate and install the SyncPack in PowerFlow. For more information, see the following topic.


**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






**WARNING:** If you are *upgrading* to this version of the SyncPack from a previous version, make a note of any settings you made on the **Configuration** pane of the various PowerFlow applications in this SyncPack, as these settings are *not* retained when you upgrade.

To activate and install a SyncPack in the PowerFlow user interface:

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

**NOTE:** If you try to activate and install a SyncPack that is already activated and installed, you can choose to "force" installation across all the nodes in the PowerFlow system.

**TIP:** If you do not see the PowerPack that you want to install, click the Filter icon (  ) on the **SyncPacks** page and select *Toggle Inactive SyncPacks* to see a list of the imported PowerPacks.

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" application appears, and you can view the Step Log for the steps. For a failed installation, the **Error Logs** window appears.
4. If you have other versions of the same SyncPack on your PowerFlow system, you can click the **[Actions]** button (  ) for that SyncPack and select *Change active version* to activate a different version other than the version that is currently running.

## Configuring Applications for the Change Management SyncPack

---

### Overview

This chapter describes how to configure PowerFlow application for the "ServiceNow Change Management" SyncPack. You can use this SyncPack to place SL1 devices into and out of maintenance mode.

This chapter covers the following topics:

<i>Creating and Aligning a Configuration Object</i> .....	14
<i>Syncing Device Maintenance from ServiceNow to SL1</i> .....	19
<i>Scheduling Device Maintenance (No MID Server)</i> .....	21
<i>Using a MID Server to Place a Device Directly in Maintenance or Schedule a Maintenance Window</i> .....	26
<i>ServiceNow Registered Events</i> .....	29

---

## Creating and Aligning a Configuration Object

A **configuration object** supplies the login credentials and other required information needed to execute the steps for an application in PowerFlow. The **Configurations** page (⚙️) of the PowerFlow user interface lists all available configuration objects for that system.

You can create as many configuration objects as you need. A PowerFlow application can only use one configuration object at a time, but you can use (or "align") the same configuration object with multiple applications.

To use this SyncPack, you will need to create one or more configuration objects in the PowerFlow user interface and align that configuration object to the applications that let you sync data between SL1 and ServiceNow.

**NOTE:** Depending on your SL1 and ServiceNow environments, you might be able to use the same configuration object with other ServiceNow SyncPacks.

**NOTE:** When either multiple SL1 stacks or multiple ServiceNow systems are involved with PowerFlow, you should create an individual configuration object for each SL1 stack or ServiceNow system. Next, create an individual schedule for each configuration object. Each schedule should use a configuration object that is specific to that single SL1 stack or ServiceNow system. Creating copies of a PowerFlow application from a SyncPack for the purpose of distinguishing between domains is not supported, and will result in issues on upgrades.

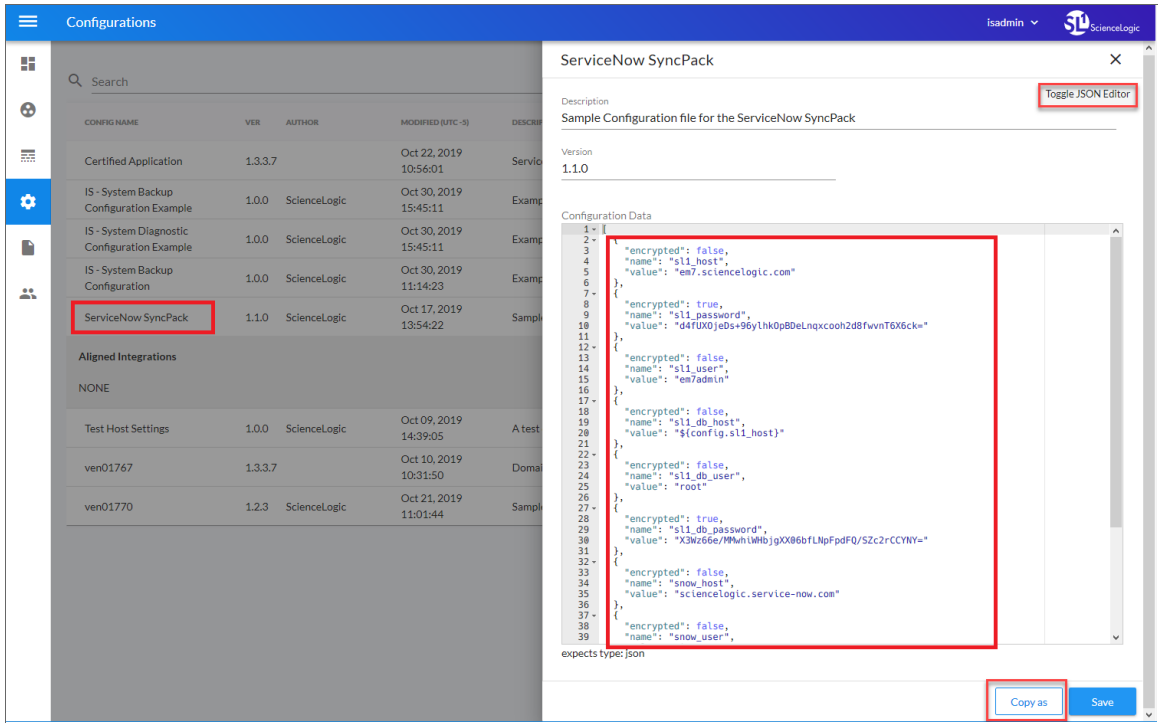
## Creating a Configuration Object

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.

**TIP:** The "ServiceNow SyncPack" configuration object contains all of the required variables. Make a copy of the configuration object and update the variables from that object to match your SL1 and ServiceNow settings.

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

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



**TIP:** Click **[Toggle JSON Editor]** to show the JSON code. Click the button again to see the fields.

3. Click **[Copy as]**. The **Create Configuration** pane appears.

**TIP:** This step is required. Do *not* use the original configuration object to run PowerFlow applications.

4. 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. In the **Configuration Data** field, include the required block of code to ensure that the applications aligned to this configuration object do not fail:

```
{
  "encrypted": false,
  "name": "<sl1_db_host>",
  "value": "${<config.sl1_host>}"
}
```

For example:

```
{
  "encrypted": false,
  "name": "sl1_db_host",
  "value": "10.2.11.42"
}
```

**TIP:** If you are using IPv6 for IP addresses, wrap the IP string in brackets, such as **https://[2001:db8:3333:4444:5555:6666:7777:8888]**

**TIP:** Click **[Toggle JSON Editor]** to show the JSON code. Click the button again to see the fields. You can also click **[Add Value]** and add a new name-value pair in the **Configuration Data Values** section.

**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.

6. In the **Configuration Data Values** field, update the default variable definitions to match your PowerFlow configuration.

**NOTE:** The **region** value is a user-defined variable that identifies your SL1 instance within ServiceNow.

7. To create a configuration variable in the JSON Editor, 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, PowerFlow 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 PowerFlow 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.

8. If you want to use OAuth2 for authentication with ServiceNow, complete the following Configuration Data Values fields:

- **snow\_oauth\_client\_id**. Enter the OAuth2 Client ID from ServiceNow.
- **snow\_oauth\_client\_secret**. Enter the OAuth2 Client secret from ServiceNow.
- **snow\_oauth\_token\_url**. Enter the full authentication URL, including host and protocol from ServiceNow. For example, "https://<test-instance-name>.service-now.com/oauth\_token.do"
- **snow\_auth\_method**: Enter *oauth* or *http\_basic*. If no value is provided, *http\_basic* will be used for connection.

**NOTE:** The configuration options listed above are included by default with the sample configuration object provided in the "ServiceNow Base" SyncPack. The configuration options are only required in the configuration object if you plan to use OAuth2 to authenticate. If the values are not present in the configuration object, normal "http\_basic" authentication will be used.

9. Click **[Save]**. You can now align this configuration object with one or more applications.

## Aligning a Configuration Object

Before you can run the applications in this SyncPack, you must first "align" a configuration object with the application you want to use.

To align a configuration object with an application:

1. From the **Applications** page of the PowerFlow user interface, open the relevant application and click **[Configure]**. The **Configurations** pane for that application appears.
2. From the **Configurations** drop-down, select the configuration object you want to use.
3. Click **[Save]** to align that configuration with the application.
4. Repeat this process for every other application you want to use.

**NOTE:** The values for the parameters that display in the **Configuration** pane with a padlock icon (🔒) are populated by the values in the configuration object.

## Creating an OAuth2 Credential Record in ServiceNow

In order to use OAuth2 for authentication with ServiceNow, you must create an OAuth2 credential record in ServiceNow. To configure the ServiceNow credential that will be used by the Connector Instance:

1. Navigate to **System OAuth > Application Registry**. The **Application Registries** page appears.
2. Click **[New]**.
3. Select **Create an OAuth API endpoint for external clients**. A new record appears.
4. Complete the following fields on the new record:
  - **Name**. Type a unique name for the credential. Required.
  - **Client ID**. The Client ID is automatically generated by the ServiceNow OAuth server.
  - **Client Secret**. Leave this empty. ServiceNow will auto-generate this when the record is saved.
  - **Refresh Token Lifespan**. Enter the length of time in seconds the Refresh Token will be valid.
  - **Access Token Lifespan**. Type the length of time in seconds that the Access Token will be valid. ScienceLogic recommends setting this to 3,600 seconds to avoid known issues for longer ServiceNow REST interactions.

**NOTE:** In a scenario where the **[Access Token Lifespan]** value is shorter than the duration of a PowerFlow step that makes multiple REST interactions with ServiceNow, the access token will expire and need to be refreshed. As a result, retries were added to several PowerFlow steps where this issue may occur. This issue will hopefully be addressed in future versions of the Base Steps SyncPack.

5. Under the **Auth Scope** section at the bottom of the page, double click **Insert a new row**.
6. In the search box that appears, click the magnifying glass icon, select the *useraccount* record, and click the checkmark icon to save.
7. Click **[Submit]** to save the new record.

Application Registries  
PowerFlow Token View: Default\*

OAuth client application details.

- Name:** A unique name.
- Client ID:** Client ID automatically generated by ServiceNow OAuth server.
- Client Secret:** Client secret for the OAuth application. Leave it empty for auto-generation.
- Refresh Token Lifespan:** Time in seconds the Refresh Token will be valid.
- Access Token Lifespan:** Time in seconds the Access Token will be valid.
- Redirect URL:** The redirect URLs authorization server redirect to. They must be absolute URLs and they are comma separated.
- Enforce Token Restriction:** Restricts the access token usage to the API 's defined in the REST API Access Policies. Unselecting this option would allow access token usage across other REST API's. [Learn more.](#)

More Info

\* Name: PowerFlow Token  
 \* Client ID: 85abad6bed82c2507f0168b91a1c1bcf  
 Client Secret: [Redacted]  
 Redirect URL: [Redacted]  
 Logo URL: [Redacted]  
 Public Client:   
 Comments: [Text Area]

Application: Global  
 Accessible from: All application scopes  
 Active:   
 \* Refresh Token Lifespan: 8,640,000  
 \* Access Token Lifespan: 3,600  
 Login URL: [Text Field]

Auth Scopes

Auth Scope
<input checked="" type="checkbox"/> useraccount
<input type="checkbox"/> Insert a new row...

## Syncing Device Maintenance from ServiceNow to SL1


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

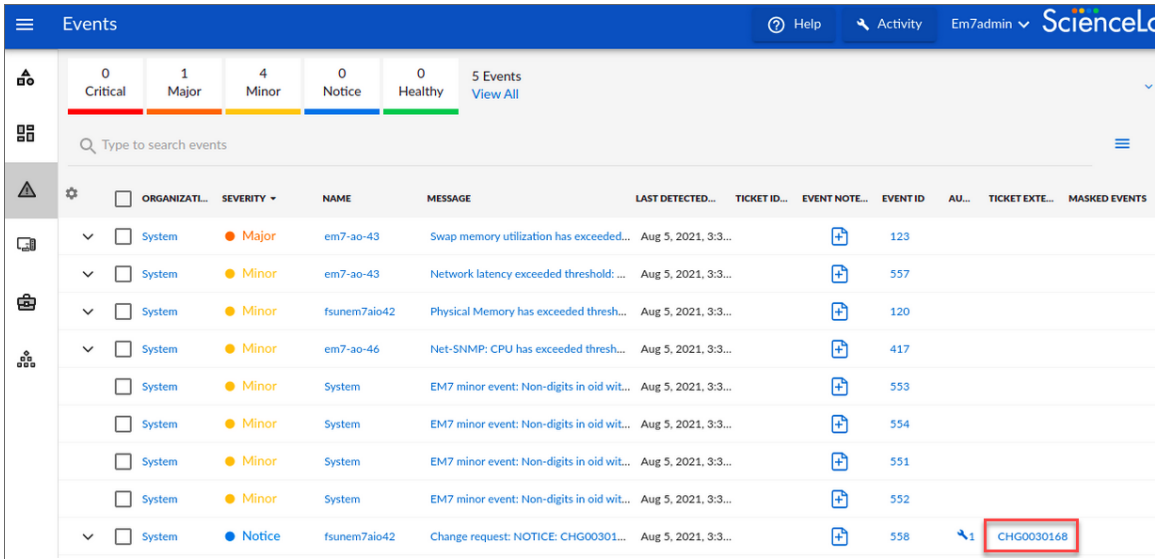
- Use the "Sync Maintenance Schedules from ServiceNow to SL1" PowerFlow application if you are *not* using a MID Server, and you want to create a change request in ServiceNow to perform *scheduled* maintenance through a maintenance window in ServiceNow. For more information, see [Scheduling Device Maintenance \(No MID Server\)](#).
- If you are looking to leverage a ServiceNow Management, Instrumentation, and Discovery (MID) Server, there are two separate ways of placing a Device in maintenance:
  - Use the "Create or Update Maintenance Schedule from ServiceNow Trigger" application to create and cancel maintenance schedules in SL1 from a ServiceNow MID Server trigger. When triggered, the application gathers data about the change request in ServiceNow and generates an SL1 Event with a severity of Notice.
  - Use the "Trigger Device Maintenance Updates via MID Server" application if you are using a ServiceNow MID Server and you want to *immediately* enable or disable maintenance on a device.

For more information, see [Using a MID Server to Place a Device Directly in Maintenance or Schedule a Maintenance Window](#).

## Viewing SL1 Events for Change Requests

Starting with version 3.2.0 of this SyncPack, when the "Create or Update Maintenance Schedule from ServiceNow Trigger" application is triggered, the application gathers data about the change request in ServiceNow and generates an SL1 Event with a severity of Notice.

For an SL1 Event that was generated by an update to a change request in ServiceNow, you can click the link for the ticket number in the **Ticket External Reference** column on the **Events** page (or click the life-ring icon  in the classic user interface) to go back to the change request in ServiceNow:

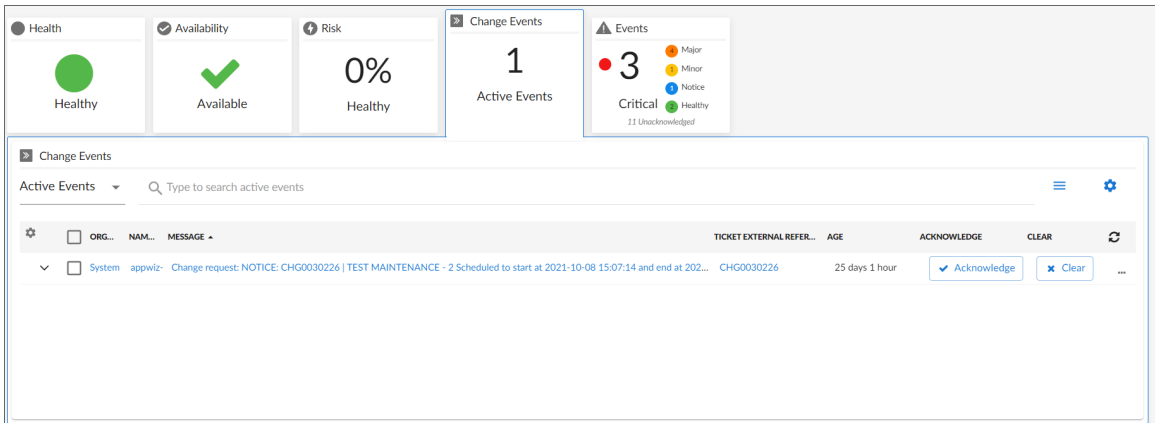


ORGANIZATI...	SEVERITY	NAME	MESSAGE	LAST DETECTED...	TICKET ID...	EVENT NOTE...	EVENT ID	AU...	TICKET EXTE...	MASKED EVENTS
System	Major	em7-ao-43	Swap memory utilization has exceeded...	Aug 5, 2021, 3:3...			123			
System	Minor	em7-ao-43	Network latency exceeded threshold: ...	Aug 5, 2021, 3:3...			557			
System	Minor	fsunem7aio42	Physical Memory has exceeded thresh...	Aug 5, 2021, 3:3...			120			
System	Minor	em7-ao-46	Net-SNMP: CPU has exceeded thresh...	Aug 5, 2021, 3:3...			417			
System	Minor	System	EM7 minor event: Non-digits in oid wit...	Aug 5, 2021, 3:3...			553			
System	Minor	System	EM7 minor event: Non-digits in oid wit...	Aug 5, 2021, 3:3...			554			
System	Minor	System	EM7 minor event: Non-digits in oid wit...	Aug 5, 2021, 3:3...			551			
System	Minor	System	EM7 minor event: Non-digits in oid wit...	Aug 5, 2021, 3:3...			552			
System	Notice	fsunem7aio42	Change request: NOTICE: CHG00301...	Aug 5, 2021, 3:3...			558		CHG0030168	

For more information about SL1 events, see the **Events** manual.

## Viewing Change Events for SL1 Business Services

If you are using SL1 version 11.1.0 or later with the Change Events feature enabled, and the device aligned with the event is in an SL1 Business Service, the event will display on the **[Change Events]** tab for that Business Service in SL1:



ORG...	NAM...	MESSAGE	TICKET EXTERNAL REFER...	AGE	ACKNOWLEDGE	CLEAR
System	appwiz:	Change request: NOTICE: CHG0030226   TEST MAINTENANCE - 2 Scheduled to start at 2021-10-08 15:07:14 and end at 202...	CHG0030226	25 days 1 hour	Acknowledge	Clear

The **Change Events** tab displays a list of events that are created when PowerFlow pulls change data from ServiceNow, including both active and cleared change events.

For more information about SL1 Business Services, see the **Business Services** manual.

---

## Scheduling Device Maintenance (No MID Server)

You can create a change request in ServiceNow to perform *scheduled* maintenance through a maintenance window in ServiceNow. You use the "Sync Maintenance Schedules from ServiceNow to SL1" PowerFlow application to sync maintenance windows from ServiceNow change requests (CHG)s to SL1 devices to place the synced devices into maintenance mode for the scheduled change window.

This process does *not* require the use of a ServiceNow MID Server.

**WARNING:** PowerFlow 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 using the *ServiceNow CMDB SyncPack*.

When you create a change request (CHG) in ServiceNow and sync it to SL1, the scheduled maintenance that SL1 creates has collections disabled by default. This is not configurable.

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

**NOTE:** If you update the scheduled times in a ServiceNow change request, you will need to cancel the change request, which also cancels the maintenance window in SL1. You will then need to create a new change request with the new time window and sync that change request to SL1.

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 'Number' field is highlighted with a red box and contains the value 'CHG0030004'. The form includes various fields for configuration, such as 'Requested by', 'Type', 'State', 'Category', 'Configuration item', 'Priority', 'Risk', 'Impact', 'Conflict status', 'Assignment group', 'Short description', 'Description', 'Justification', and 'Implementation plan'. The 'Number' field is highlighted with a red box.

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

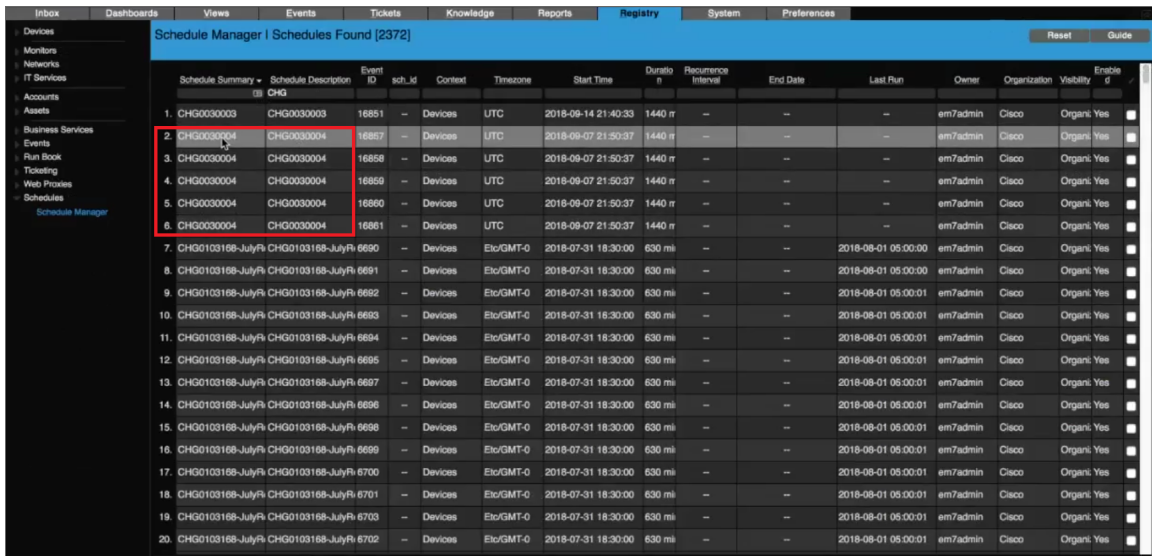
Configuration Item	Class
<input type="checkbox"/> <a href="#">gdlb-test0-cu3-34-6</a>	<a href="#">Linux Server</a>

9. In the PowerFlow user interface, go to the **Applications** page and select the "Sync Maintenance Schedules from ServiceNow to SL1" application.
10. Click **[Configure]** . The **Configuration** pane appears:


11. As needed, update the following options from the **Configuration** pane:
  - **Configuration**. Select the relevant configuration object to align with this application. You cannot edit fields that are populated by the configuration object. Required.
  - **New\_Change\_Request\_State**. The State ID from ServiceNow of the scheduled change request that this application accesses to pull to schedule maintenance windows in SL1. The default is -2.
  - **read\_timeout**. Specify the maximum amount of time in seconds the application should wait for a response before timing out. The default is 20 seconds.
  - **Canceled\_Change\_Request\_State**: The State ID for a canceled change request that this application accesses to pull to schedule maintenance windows in SL1. The default is 4.
  - **New\_Change\_Task\_State**: The State ID of the scheduled change task that this application accesses to pull to cancel maintenance windows in SL1. The default is 1.
  - **Canceled\_Change\_Task\_State**: The State ID for a canceled change task that this application accesses to pull to cancel maintenance windows in SL1. The default is 4.



- **Process\_Change\_Tasks**: Select this option to enable change task processing, which allows the request to be broken down into Change Tasks, which results in two extra pulls of the data. The default is unselected.
  - **collection\_polling**. When selected, leaves synced devices in a collection enabled state after a synced schedule completes. The devices will be put in maintenance but the devices will remain enabled for polling. By default, the devices are left in a collection disabled state.
  - **patch\_window**. Specifies a patch window from 0 to 60 minutes.
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 **[Save]**. The **Configuration** pane automatically closes after this message appears.
  14. Click **[Run]**  to run the application.
  15. While the "Sync Maintenance Schedules from ServiceNow to SL1" 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.
  16. After the "Sync Maintenance from ServiceNow to SL1" 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 hr	--	--	--	em7admin	Cisco	Organic	Yes
2. CHG0030004	CHG0030004	16857	--	Devices	UTC	2018-09-07 21:50:37	1440 hr	--	--	--	em7admin	Cisco	Organic	Yes
3. CHG0030004	CHG0030004	16858	--	Devices	UTC	2018-09-07 21:50:37	1440 hr	--	--	--	em7admin	Cisco	Organic	Yes
4. CHG0030004	CHG0030004	16859	--	Devices	UTC	2018-09-07 21:50:37	1440 hr	--	--	--	em7admin	Cisco	Organic	Yes
5. CHG0030004	CHG0030004	16860	--	Devices	UTC	2018-09-07 21:50:37	1440 hr	--	--	--	em7admin	Cisco	Organic	Yes
6. CHG0030004	CHG0030004	16861	--	Devices	UTC	2018-09-07 21:50:37	1440 hr	--	--	--	em7admin	Cisco	Organic	Yes
7. CHG0103168-JulyR	CHG0103168-JulyR	6690	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:00	em7admin	Cisco	Organic	Yes
8. CHG0103168-JulyR	CHG0103168-JulyR	6691	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:00	em7admin	Cisco	Organic	Yes
9. CHG0103168-JulyR	CHG0103168-JulyR	6692	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
10. CHG0103168-JulyR	CHG0103168-JulyR	6693	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
11. CHG0103168-JulyR	CHG0103168-JulyR	6694	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
12. CHG0103168-JulyR	CHG0103168-JulyR	6695	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
13. CHG0103168-JulyR	CHG0103168-JulyR	6697	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
14. CHG0103168-JulyR	CHG0103168-JulyR	6696	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
15. CHG0103168-JulyR	CHG0103168-JulyR	6698	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
16. CHG0103168-JulyR	CHG0103168-JulyR	6699	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
17. CHG0103168-JulyR	CHG0103168-JulyR	6700	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
18. CHG0103168-JulyR	CHG0103168-JulyR	6701	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
19. CHG0103168-JulyR	CHG0103168-JulyR	6703	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes
20. CHG0103168-JulyR	CHG0103168-JulyR	6702	--	Devices	Etc/GMT-0	2018-07-31 18:30:00	630 ms	--	--	2018-08-01 05:00:01	em7admin	Cisco	Organic	Yes

17. 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**.
18. 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.
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" application runs, the application cancels that maintenance sync.

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

---

## Using a MID Server to Place a Device Directly in Maintenance or Schedule a Maintenance Window

You can use this SyncPack with a ServiceNow MID Server for the following scenarios:


- Placing a device directly in maintenance.
- Scheduling a maintenance window

To set up either scenario, perform the following:

1. [Configure and run the "Sync Change Management Requirements" application](#) in PowerFlow.
2. [Create a ServiceNow Credential and Credential Affinity in ServiceNow](#), or [Configure ServiceNow to use an API Key for PowerFlow Authentication](#)

## Configuring and Running the "Sync Change Management Requirements" Application in PowerFlow

Prerequisites for the MID Server workflow:

1. In PowerFlow, open the "Sync Change Management Requirements" application and click the **[Configure]** button (). The **Configuration** pane appears.
2. Update the following fields:
  - **Configuration**. Select the configuration object with the saved credentials for ServiceNow and SL1.
  - **is\_host**. Specify the IP of the cluster VIP
3. Click **[Save]**.
4. Click the **[Run]** button. The application syncs the configuration objects from PowerFlow to ServiceNow so that ServiceNow can tell PowerFlow to which SL1 system it needs to send the change request.
5. In ServiceNow, create a credential for the sync. For more information, see [Creating a ServiceNow Credential and Credential Affinity](#) or [Configure ServiceNow to use an API Key for PowerFlow Authentication](#).

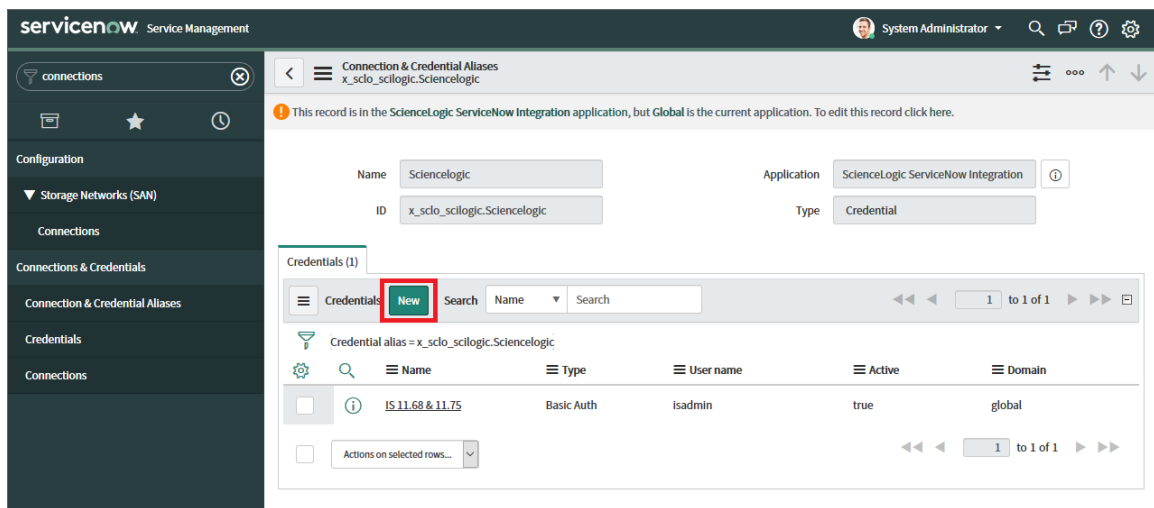
**NOTE:** You do *not* need to run or schedule the "Create or Update Maintenance Schedule from ServiceNow Trigger" application or the "Trigger Device Maintenance Updates via MID Server" application in PowerFlow. Those applications are triggered by the MID Server, which is triggered first by a registered event in ServiceNow Event Management. For more information about registered events, including examples of other triggering events you can define in ServiceNow, see [ServiceNow Registered Events](#).

## Creating a ServiceNow Credential and Credential Affinity

Before you can set up maintenance sync with the MID Server in ServiceNow, you need to create a credential for the MID Server. You should have access to the "Integration Services" section of the **Discovery Dependents** page (`x_sclo_scilogic_discovery_dependent`) in ServiceNow, which is provided by the "Integration Services" import set.

To create a credential to connect to PowerFlow:

1. In ServiceNow, go to **Connections & Credentials > Connection & Credential Aliases**.
2. From the **Connection & Credential Aliases** list, select **ScienceLogic**. 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 PowerFlow 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 PowerFlow 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]**.

## Configuring ServiceNow to use an API Key for PowerFlow Authentication

If you want to use an API key for authentication instead of using the default "isadmin" user, you will need to add an HTTP header to the **Integration Services Application Run** record in ServiceNow.

1. In PowerFlow, create an API key on the **API Keys** page and save the key for step 9, below.
2. In ServiceNow, go to **System Web Services > Outbound > REST Message** and search for *Integration Services Application Run*.
3. Select the **Integration Services Application Run** record that is connected to the *ScienceLogic SL1: Service Catalog Request Integration* in the **Application** column. The **Integration Services Application Run** record appears:

The screenshot shows the ServiceNow interface for editing a REST Message record. The record name is "Integration Services Application Run" and it is associated with the application "ScienceLogic SL1: Service Catalog Request Integration". The endpoint is "https://\$[ip]/api/v1/applications/run". The authentication method is "HTTP Request". Under the "HTTP Headers" section, a header with the name "content-type" and value "application/json" is defined. Below this, a table lists the HTTP methods for this record, showing a single entry for "IS Application Run" using the "POST" method at the same endpoint.

Name	Value
content-type	application/json

Name	HTTP method	Endpoint
IS Application Run	POST	https://\$[ip]/api/v1/applications/run

4. In the **HTTP Methods** section, click **IS Application Run**.
5. Click the **[HTTP Request]** tab.
6. If needed, click *Insert* from the **Additional actions** menu at the top left of the record.
7. In the **Use MID Server** field, select the MID server you want to use.
8. In the **HTTP Headers** section, type *PF-APIKEY* in the **Name** field.
9. In the **HTTP Headers** section, paste the API key from step 1 in the **Value** field.
10. Click the **[Update]** button in the lower left of the window.

---

## ServiceNow Registered Events

This section describes the commands that you can use to generate registered events in ServiceNow. These events can trigger actions in PowerFlow, such as specifying one or more CIs for monitoring, or putting a CI into maintenance.

**WARNING:** This section is recommended for advanced ServiceNow administrators.

These events use the `gs.eventQueue` command, using the following format:

```
eventQueue(String name, Object instance, String parm1, String parm2)
```

For more information about registered events, see the ServiceNow documentation:

[https://developer.servicenow.com/dev.do#!/learn/learning-plans/quebec/new\\_to\\_servicenow/app\\_store\\_learnv2\\_automatingapps\\_quebec\\_what\\_are\\_events](https://developer.servicenow.com/dev.do#!/learn/learning-plans/quebec/new_to_servicenow/app_store_learnv2_automatingapps_quebec_what_are_events).

## Change Request Events

The following events can be triggered to queue to the event manager. The actual trigger for the following events is not included in the application, but you can configure a custom trigger. An example is provided in the [Change Request Events Example](#), below.

## Direct to Device: `x_sclo_scilogic.device_maintenance`

The "Trigger Device Maintenance Updates via MID Server" application receives a group of one or more synced devices from the ServiceNow 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.

### Command

```
gs.eventQueue('x_sclo_scilogic.device_maintenance',current, action, affected_ci);
```

## Event Fields

Field	Description
<code>x_sclo_scilogic.device_maintenance</code>	Unique name of the event.
<code>current</code>	The table to which the event applies.
<code>action</code>	Parm1: An array that includes the action to be performed ( <code>enable_maintenance</code> or <code>disable_maintenance</code> ) and the <code>sys_id</code> of the task. The task is not required; the action is required.
<code>affected_ci</code>	Parm2: An array of device <code>sys_ids</code> that need to be enable or disabled for maintenance mode.

## Scheduled Device Maintenance: `x_sclo_scilogic.device_maintenance_skd`

The "Create or Update Maintenance Schedule from ServiceNow Trigger" application handles scheduling and canceling maintenance schedules in SL1 from a trigger from a ServiceNow Management, Instrumentation, and Discovery (MID) Server.

The "Create or Update Maintenance Schedule from ServiceNow Trigger" receives a group of one or more synced devices from the ServiceNow Management, Instrumentation, and Discovery (MID) Server and checks for the schedule and cancel 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.

As a result, in ServiceNow you can bring change requests out of a scheduled state and update them, and all of the updates to those change requests are synced back to SL1 in real time, even if those change requests were already scheduled.

## Command

```
gs.eventQueue('x_sclo_scilogic.device_maintenance_skd', current,
'schedule', current.getUniqueValue());
```

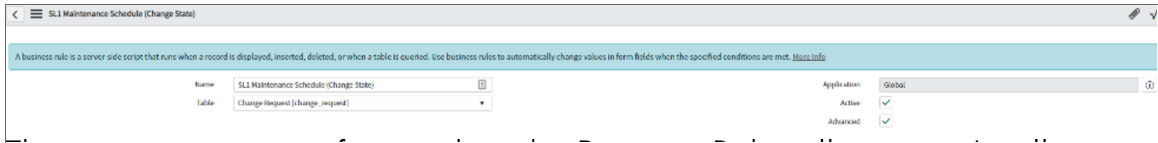
## Event Fields

Field	Description
<code>x_sclo_scilogic.device_maintenance_skd</code>	Unique name of the event.
<code>current</code>	The table to which the event applies.
<code>schedule</code>	Parm1: Accepts the following variables: <ul style="list-style-type: none"> <li><code>schedule</code>: Creates a schedule in SL1.</li> <li><code>cancel</code>: Deletes any currently scheduled maintenance.</li> <li><code>event</code>: Creates an event in SL1.</li> </ul>
<code>current.getUniqueValue()</code>	Parm2: The <code>sys_id</code> of the change request.

## Change Request Events Example

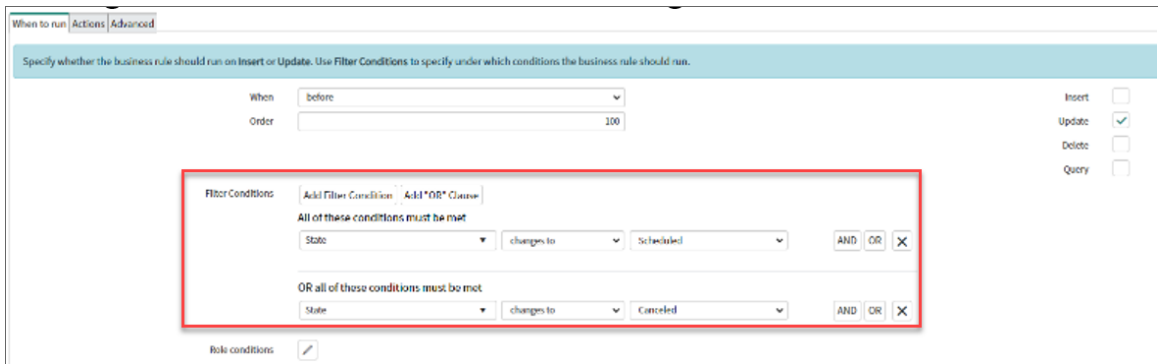
There are many different ways to utilize the registered events that are supplied in the certified application. Below is one possible way of triggering by using business rule on the change request table.

1. Navigate to the change request table and create a business rule:



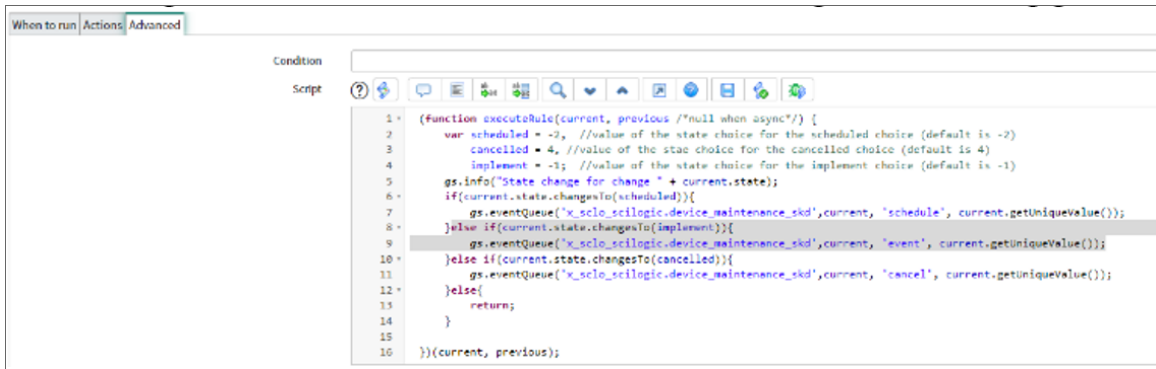
The screenshot shows the 'Business Rule' configuration interface. The title bar reads 'SLI Maintenance Schedule (Change State)'. Below the title, there is a descriptive text: 'A business rule is a server side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. [More info](#)'. The configuration fields are: Name: 'SLI Maintenance Schedule (Change State)', Table: 'Change Request (change\_request)', Application: 'Global', Active: , and Advanced: .

2. Configure the Business Rule to trigger when the state of the change request changes to *Scheduled* or *Canceled*:



The screenshot shows the 'When to run' configuration page. The title bar reads 'When to run | Actions | Advanced'. Below the title, there is a descriptive text: 'Specify whether the business rule should run on Insert or Update. Use Filter Conditions to specify under which conditions the business rule should run.' The configuration fields are: When: 'before', Order: '100', Insert: , Update: , Delete: , and Query: . The 'Filter Conditions' section is highlighted with a red box and contains two conditions: 'All of these conditions must be met' with 'State changes to Scheduled' and 'OR all of these conditions must be met' with 'State changes to Canceled'.

3. Click the **[Advanced]** tab, where you can configure the trigger for the event queue:



This script block is set up to trigger three different times, depending on the state change:

```
(function executeRule(current, previous /*null when async*/) {
    var scheduled = -2, //value of the state choice for the scheduled
choice (default is -2)
        cancelled = 4, //value of the state choice for the cancelled choice
(default is 4)
        implement = -1; //value of the state choice for the implement
choice (default is -1)
    gs.info("State change for change " + current.state);
    if(current.state.changesTo(scheduled)){
        gs.eventQueue('x_sclo_scilogic.device_maintenance_skd',current,
'schedule', current.getUniqueValue());
    }else if(current.state.changesTo(implement)){
        gs.eventQueue('x_sclo_scilogic.device_maintenance_skd',current,
'event', current.getUniqueValue());
    }else if(current.state.changesTo(cancelled)){
        gs.eventQueue('x_sclo_scilogic.device_maintenance_skd',current,
'cancel', current.getUniqueValue());
    }else{
        return;
    }
}) (current, previous);
```



---

# Chapter

# 4

## Troubleshooting the Change Management SyncPack

---

### Overview

This chapter includes troubleshooting resources and procedures to use with the "ServiceNow Change Management" SyncPack.

This chapter covers the following topics:

<i>Initial Troubleshooting Steps</i> .....	34
<i>Resources for Troubleshooting</i> .....	34
<i>Allowing Cross-Scoped Access in ServiceNow</i> .....	38

---

## Initial Troubleshooting Steps

PowerFlow acts as a middle server between data platforms. For this reason, the first steps should always be to ensure that there are no issues with the data platforms with which PowerFlow is talking. There might be additional configurations or actions enabled on ServiceNow or SL1 that result in unexpected behavior. For detailed information about how to perform the steps below, see [Resources for Troubleshooting](#).

### SL1 PowerFlow

1. Run `docker service ls` on the PowerFlow server:
  - Note the Docker container version.
  - Verify that the Docker services are running.
2. If a certain service is failing, make a note of the service name and version.
3. If a certain service is failing, run `docker service ps <service_name>` to see the historical state of the service and make a note of this information. For example: `docker service ps iservices_contentapi`.
4. Make a note of any logs impacting the service by running `docker service logs <service_name>`. For example: `docker service logs iservices_couchbase`.

### ServiceNow

1. Make a note of the ServiceNow version and SyncPack version, if applicable.
2. Make a note if you are running a ServiceNow certified application or a Service Graph SyncPack.
3. Make a note of the SyncPack application that is failing in PowerFlow.
4. Make a note of what step is failing in the application, try running the application in debug mode, and capture any traceback or error messages that occur in the step log.

---

## Resources for Troubleshooting

This section contains port information for PowerFlow and troubleshooting commands for Docker, Couchbase, and the PowerFlow API.

### Useful PowerFlow Ports

- **`https://<IP of PowerFlow>:8091`**. Provides access to Couchbase, a NoSQL database for storage and data retrieval.
- **`https://<IP of PowerFlow>:15672`**. Provides access to the RabbitMQ Dashboard, which you can use to monitor the service that distributes tasks to be executed by PowerFlow workers.
- **`https://<IP of PowerFlow>/flower/dashboard`**. Provides access to Flower, a tool for monitoring and administrating Celery clusters.

- <https://<IP of PowerFlow>:3141>. Provides access to the pypiserver service. which you can use to see if SyncPacks have been correctly uploaded to Devpi container.

**IMPORTANT:** Port 5556 must be open for both PowerFlow and the client.

## Helpful Docker Commands

PowerFlow is a set of services that are containerized using Docker. For more information about Docker, see the [Docker tutorial](#).

Use the following Docker commands for troubleshooting and diagnosing issues with PowerFlow:

### Viewing Container Versions and Status

To view the PowerFlow version, SSH to your instance and run the following command:

```
rpm -qa | grep powerflow
```

To view the individual services with their respective image versions, SSH to your PowerFlow instance and run the following command:

```
docker service ls
```

In the results, you can see the container ID, name, mode, status (see the *replicas* column), and version (see the *image* column) for all the services that make up PowerFlow:

```
[root@fsunis4lab ~]# docker service ls
ID                NAME                MODE                REPLICAS                IMAGE                PORTS
mm1hg3v301       iservices_gui        replicated           2/1                     repository.auto.sciencelogic.local:5000/is-gui:1.7.0        *780->80/tcp,*443->443/tcp
40w91t1vmb3      iservices_redis      replicated           1/1                     redis:4.0.2
jlm6h1j2umf      iservices_flower     replicated           1/1                     repository.auto.sciencelogic.local:5000/is-worker:1.7.0        *:5555->5555/tcp
lh3pt2l91zsf     iservices_scheduler replicated           1/1                     repository.auto.sciencelogic.local:5000/is-worker:1.7.0
htmltv6xhx       iservices_contentapi replicated           1/1                     repository.auto.sciencelogic.local:5000/is-api:1.7.0          *:5000->5000/tcp
pyln5qgsudm1     iservices_rabbitmq   replicated           1/1                     rabbitmq:3
x1l2m83j3s46     iservices_visual     replicated           2/1                     dockersamples/visualizer:latest                            *:8081->8080/tcp
vcy38w8ouaw      iservices_couchbase  replicated           1/1                     repository.auto.sciencelogic.local:5000/is-couchbase:1.7.0    *:8091->8091/tcp,*8092->8092/tcp
0->8093/tcp,*8094->8094/tcp,*11210->11210/tcp
z1bxatxz7uf      iservices_steprunner replicated           5/5                     repository.auto.sciencelogic.local:5000/is-worker:1.7.0
```

### Restarting a Service

Run the following command to restart a single service:

```
docker service update --force <service_name>
```

### Stopping all PowerFlow Services

Run the following command to stop all PowerFlow services:

```
docker stack rm iservices
```

### Restarting Docker

Run the following command to restart Docker:

```
systemctl restart docker
```

**NOTE:** Restarting Docker does not clear the queue.

## Diagnosis Tools

Multiple diagnosis tools exist to assist in troubleshooting issues with the PowerFlow platform:

- **Docker PowerPack.** This PowerPack monitors your Linux-based PowerFlow server with SSH (the PowerFlow ISO is built on top of an Oracle Linux Operating System). This PowerPack provides key performance indicators about how your PowerFlow server is performing. For more information on the Docker PowerPack and other PowerPacks that you can use to monitor PowerFlow, see the "Using SL1 to Monitor SL1 PowerFlow" chapter in the *SL1 PowerFlow Platform* manual.
- **Flower.** This web interface tool can be found at the /flower endpoint. It provides a dashboard displaying the number of tasks in various states as well as an overview of the state of each worker. This tool shows the current number of active, processed, failed, succeeded, and retried tasks on the PowerFlow platform. This tool also shows detailed information about each of the tasks that have been executed on the platform. This data includes the UUID, the state, the arguments that were passed to it, as well as the worker and the time of execution. Flower also provides a performance chart that shows the number of tasks running on each individual worker.
- **Debug Mode.** All applications can be run in "debug" mode via the PowerFlow API. Running applications in debug mode may slow down the platform, but they will result in much more detailed logging information that is helpful for troubleshooting issues. For more information on running applications in Debug Mode, see [Retrieving Additional Debug Information](#).
- **Application Logs.** All applications generate a log file specific to that application. These log files can be found at `/var/log/iservices` and each log file will match the ID of the application. These log files combine all the log messages of all previous runs of an application up to a certain point. These log files roll over and will get auto-cleared after a certain point.
- **Step Logs.** Step logs display the log output for a specific step in the application. These step logs can be accessed via the PowerFlow user interface by clicking on a step in an application and bringing up the **Step Log** tab. These step logs display just the log output for the latest run of that step.
- **Service Logs.** Each Docker service has its own log. These can be accessed via SSH by running the following command:

```
docker service logs -f <service_name>
```

## Retrieving Additional Debug Information (Debug Mode)

The logs in PowerFlow use the following **loglevel** settings, from most verbose to least verbose:

- **10.** Debug Mode.
- **20.** Informational.
- **30.** Warning. This is the default settings if you do not specify a loglevel.
- **40.** Error.

**WARNING:** If you run applications in Debug Mode ("loglevel": 10), those applications will take longer to run because of increased I/O requirements. Enabling debug logging using the following process is the only recommended method. ScienceLogic does not recommend setting "loglevel": 10 for the whole stack with the **docker-compose** file.

To run an application in Debug Mode using the PowerFlow user interface:

1. Select the PowerFlow application from the **Applications** page.
2. Hover over the **[Run]** button and select *Custom Run* from the pop-up menu. The **Custom Run** window appears.
3. Select the Logging Level. *Debug* is the most verbose and will take longer to run.
4. Specify the configuration object for the custom run in the **Configuration** field, and add any JSON parameters in the **Custom Parameters** field, if needed.
5. Click **[Run]**.

To run an application in Debug Mode using the API:

1. POST the following to the API endpoint:

```
https://<PowerFlow_IP>/api/v1/applications/run
```

2. Include the following in the request body:

```
{  
  "name": "<application_name>",  
  "params": {  
    "loglevel": 10  
  }  
}
```

After running the application in Debug Mode, review the step logs in the PowerFlow user interface to see detailed debug output for each step in the application. This information is especially helpful when trying to understand why an application or step failed:

The screenshot displays the ScienceLogic PowerFlow interface for an application named 'Delete Devices From SL1'. The workflow diagram shows a sequence of steps: 'Pull Disabled Devices from VCUG in SL1' (highlighted with a red box and an error icon), followed by two parallel steps 'Pull Affected Device Info from SL1 (GSQL)' and 'Pull Affected Device Info from SL1 (MySQL)', then 'Verify Device Delete Requests', and finally 'Delete Devices'. Below the diagram is a 'STEP LOG' table with columns for Module, Date (UTC-4), Log Level, and Message. The table shows several log entries, with the most recent one being an ERROR from the BaseStep module, which contains a detailed traceback of the failure.

Module	Date (UTC-4)	Log Level	Message
BaseStep	Aug 29, 2023 10:48:21, 347	INFO	Executing: Pull Disabled Devices from VCUG in SL1 - steps/PullAndProcessDisabledDevices.py
ipaas_logger	Aug 29, 2023 10:48:21, 348	FLOW	Start Pull Disabled Devices from VCUG in SL1
ipaas_logger	Aug 29, 2023 10:48:21, 355	FLOW	Step Pull Disabled Devices from VCUG in SL1 still failed after 0 retries
BaseStep	Aug 29, 2023 10:48:21, 357	ERROR	Traceback (most recent call last): File "/usr/local/lib/python3.8/site-packages/ipaascore/BaseStep.py", line 601, in execute_step self.retry_step(task=task, exc=err, File "/usr/local/lib/python3.8/site-packages/ipaascore/BaseStep.py", line 961, in retry_step task.retry(...) File "/usr/local/lib/python3.8/site-packages/celery/app/task.py", line 706, in retry raise_with_context(exc) File "/usr/local/lib/python3.8/site-packages/ipaascore/BaseStep.py", line 579, in execute_step self.execute() File "/usr/local/lib/python3.8/site-packages/ipaascore/BaseStep.py", line 51, in inner_execute execute = method(self) File "/var/syncpacks/virtualems/servicenow_cmdb_syncpack/lib/python3.8/site-packages/servicenow_cmdb_syncpack/Steps/PullAndProcessDisabledDevices.py", line 69, in execute raise MissingRequiredStepParameterf ipaascommon.ipaas_exceptions.MissingRequiredStepParameter: target_vcut is required but is not populated in either delete_devices or in Sync Service ...

You can also run an application in debug using curl via SSH:

1. SSH to the PowerFlow instance.
2. Run the following command:

```
curl -v -k -u isadmin:<password> -X POST "https://<your_
hostname>/api/v1/applications/run"
-H 'Content-Type: application/json' -H 'cache-control: no-cache' -d
 '{"name":
"interface_sync_sciencelogic_to_servicenow","params": {"loglevel":
10}}'
```

## Allowing Cross-Scoped Access in ServiceNow

When using custom tables, you might need to configure cross-scope access for the ScienceLogic plugin. 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.ScopeAccessNotGrantedExceptio
n: 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. For newer releases of ServiceNow, click the globe icon (🌐) to change the scope. On older releases, 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). 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:
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*. Be sure to verify the application to which the table belongs, and use that value as the target scope in this field.
  - **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 the [Cross-scope privilege record](#) topic in the ServiceNow documentation.



---

# Appendix

# A

## ServiceNow Change Management API Endpoints

---

### Overview

This appendix describes the customized ServiceNow API Endpoints that were created for the ServiceNow Change Management SyncPack. These scripted endpoints reduce the amount of REST calls that PowerFlow 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 chapter covers the following topics:

<a href="#">Change Requests</a> .....	42
---------------------------------------	----

---

# Change Requests

## HTTP Method

GET

## Resource Path

```
/api/x_sclo_scilogic/v2/sciencelogic/change_requests?record_type=change_request&state=1&region=ScienceLogic
```

## Pagination

Enabled

## Overview

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 will *only* pull Change Requests and Change Tasks that are in the future.

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\' or '\record_type\' is not defined and is required.

## Example (Request URL)

```
https://<your Instance>.service-now.com/api/x_sclo_scilogic/v2/sciencelogic/change_requests?record_type=change_request&state=-2@ion=ven70_sl151_pf199&sysparm_offset=1&sysparm_limit=1
```

## Examples (Response)

Change Request:

```
{
  "results": [
    {
      "sys_id": "129994931bf91110f219866fdc4bcbb6",
      "number": "CHG0030115",
      "type": "Normal",
      "state_value": "-2",
      "state": "Scheduled",
      "short_description": "Firmware upgrade Printer (x8)",
      "planned_start_date": "2022-09-30 17:00:00",
      "planned_end_date": "2022-09-30 17:30:00",
      "device": [
        {
          "sys_id": "75febcd21b3d99106af3dac4bd4bcb95",
          "name": "Printer: SIM-0019004",
          "class": "cmdb_ci_printer",
          "id": "488",
          "region": "ven70_sl151_pf199"
        }
      ]
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 1,
  "return_count": 1,
}
```

```
"total_count": 2
}
```

Change Task:

```
{
  "results": [
    {
      "sys_id": "af2554eb1b7d5110f219866fdc4bcbcd",
      "number": "CTASK0010073",
      "state_value": "1",
      "state": "Open",
      "short_description": "firmware upgrade",
      "planned_start_date": "2022-10-01 14:30:00",
      "planned_end_date": "2022-10-01 15:00:00",
      "device": [
        {
          "sys_id": "01fef8d21b3d99106af3dac4bd4bcb1f",
          "name": "IP Router: SIM-0019217",
          "class": "cmdb_ci_ip_router",
          "id": "439",
          "region": "ven70_sl151_pf199"
        }
      ]
    }
  ],
  "sysparm_offset": 0,
  "sysparm_limit": 1,
  "return_count": 1,
  "total_count": 1
}
```

© 2003 - 2024, 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](mailto:legal@sciencelogic.com). For more information, see <https://sciencelogic.com/company/legal>.

ScienceLogic

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010