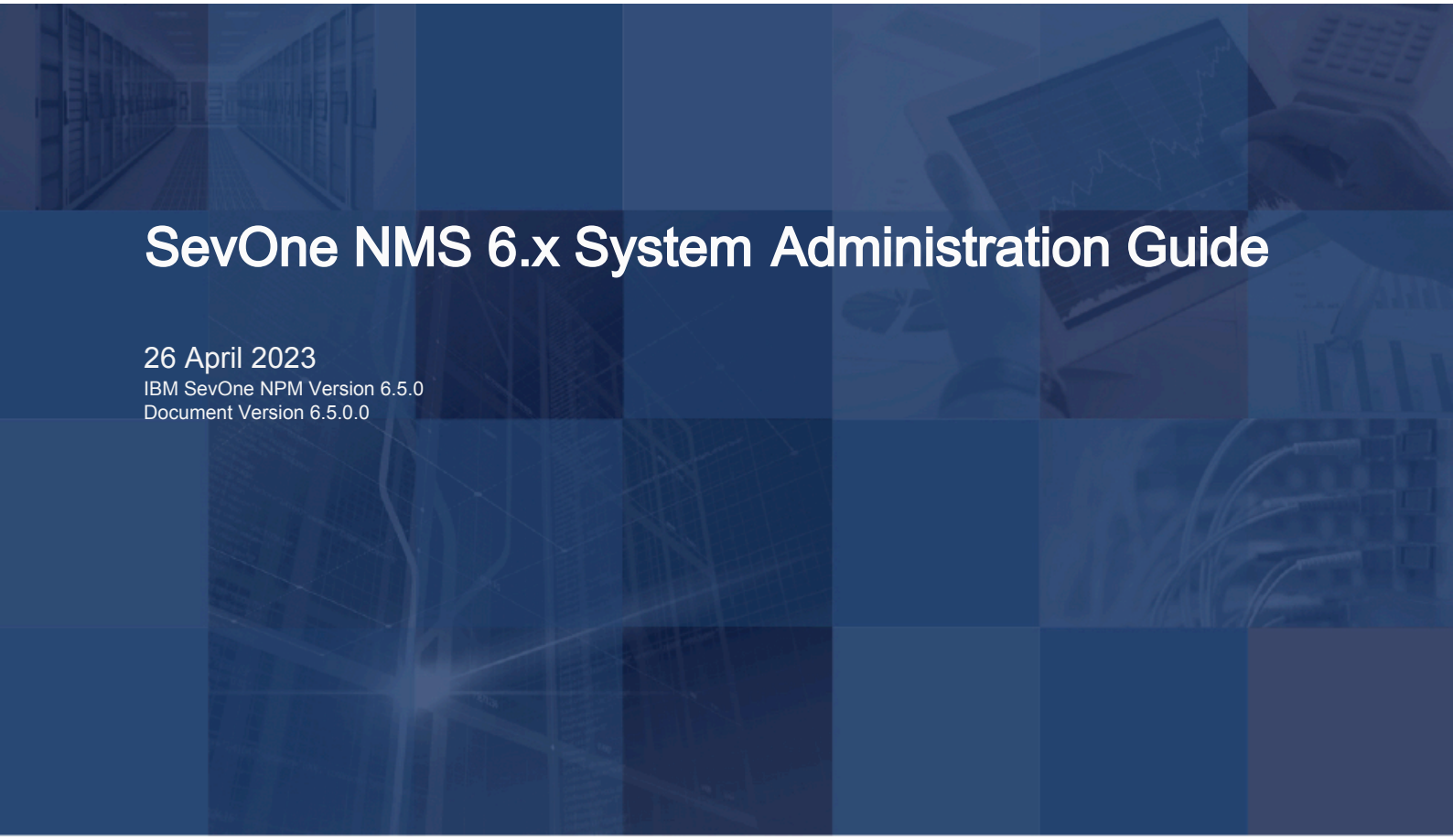


SevOne NMS 6.x System Administration Guide

26 April 2023

IBM SevOne NPM Version 6.5.0

Document Version 6.5.0.0

Table of Contents

SevOne Documentation

All documentation is available from the IBM SevOne Support customer portal.

1 About this Document

This manual describes workflows for users assigned to administrative roles that grant permission to system administration workflows. For the purposes of this document, system administration workflows include the typical initial application implementation settings, cluster level settings, settings that affect the application functionality holistically, and settings that affect all groups of things such as device types, user roles, system level passwords, etc. When you are assigned to a role that does not permit you to use a workflow, that workflow does not appear for you.

Terminology usage...

In this guide if there is,

- [any reference to *master*] OR
- [[if a CLI command contains *master*] AND/OR
- [its output contains *master*]],

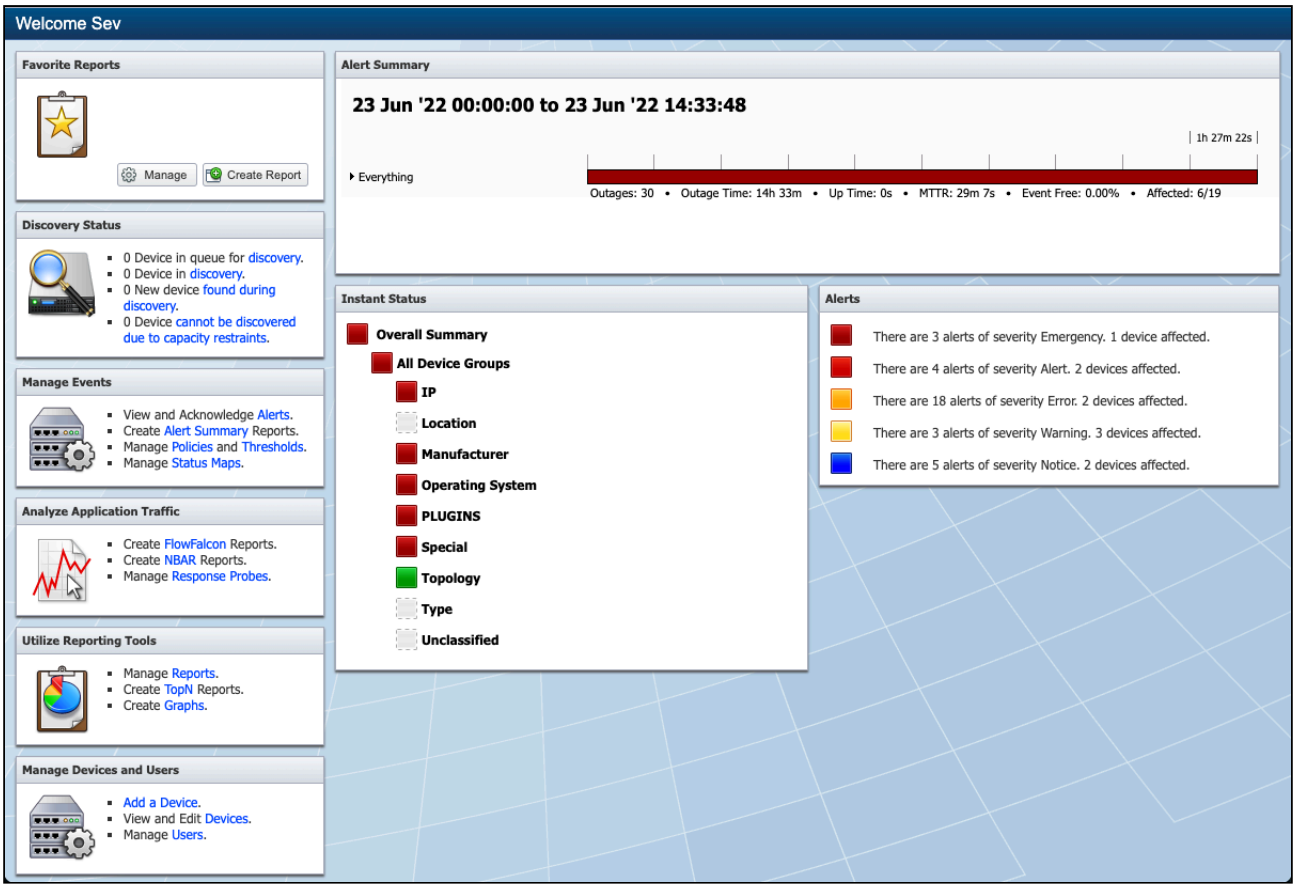
it means *leader*.

And, if there is any reference to *slave*, it means *follower*.

2 Introducing SevOne

2.1 Network Management System

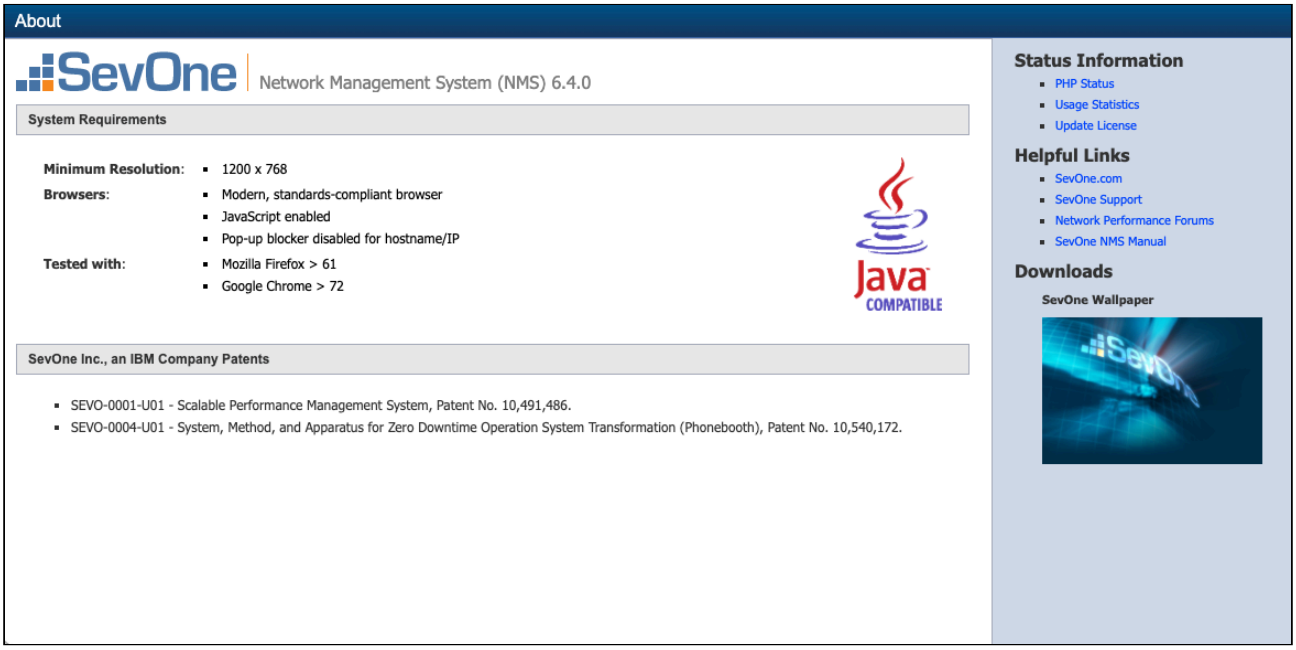
Each SevOne Performance Appliance Solution (PAS) and SevOne Dedicated NetFlow Collector (DNC) is packaged as a full-featured integrated appliance ready to monitor your network or IT operations within minutes of installation. No additional software, hardware, or external databases are needed for the SevOne PAS/DNC to operate. The SevOne Network Management System (NMS) software enables you to poll and monitor what happens in your network right now and to analyze what occurred historically.



2.2 About SevOne NMS

When you log on, the navigation bar appears across the top of the page. The navigation bar contains menus and navigation icons. User roles enable you to define which SevOne NMS workflows users can access and which devices users can access - for details on User roles, please refer to section **User Role Manager** in *SevOne NMS System Administration Guide*. The online help describes workflows for users with permissions to access to all SevOne NMS features. If your security settings do not permit you to use a workflow, that workflow does not appear for you.

On the navigation bar, click the **Administration** menu and select **About** to display the About page.



2.2.1 The About Page

The upper section displays the SevOne NMS software version number and system requirements. The lower section displays the SevOne Inc., an IBM Company Patents. The right side provides access to status information for administration users and other helpful links.

2.3 Initial SevOne NMS Implementation

The *SevOne NMS Installation Guide* provides instructions for how to rack up your SevOne appliance and to get the appliance into your network. The *SevOne NMS Implementation Guide* provides instructions for how to begin your SevOne NMS implementation.

All documentation is available from the SevOne Support website.

3 Login

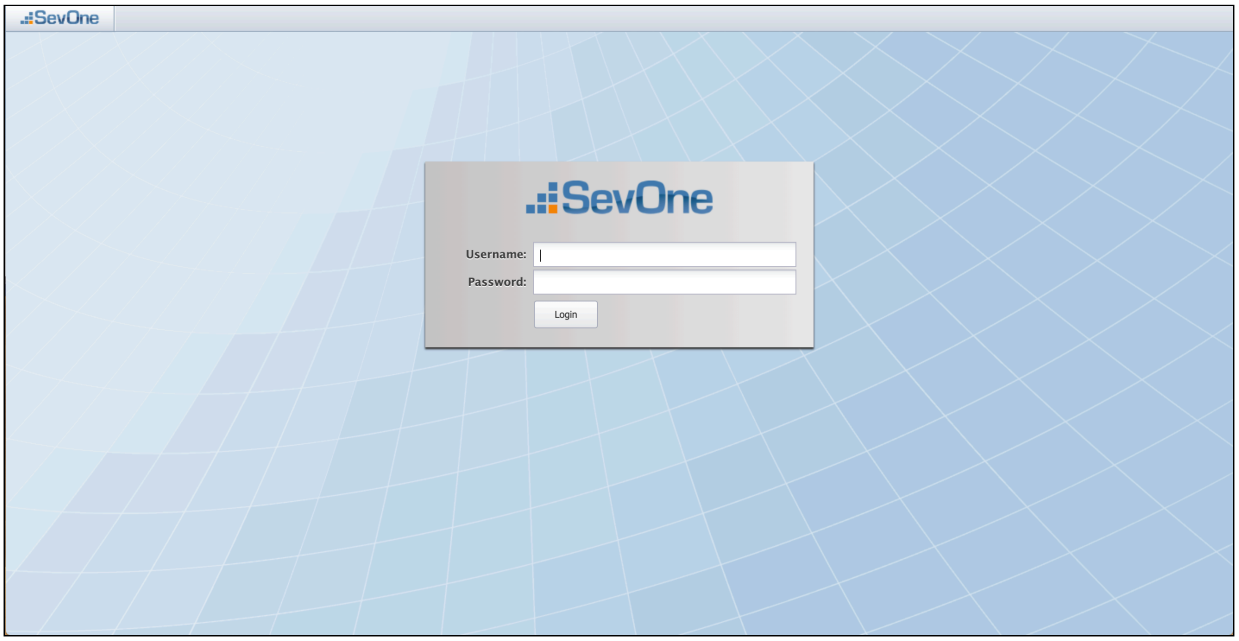
The Login page provides security. Contact your SevOne NMS user managers to create your account user name and password.

To access the Login page, navigate to the appropriate URL in your browser. Please read the following **EULA** (END-USER LICENSE AGREEMENT) terms of services.

1. for **License Agreement**, please read the international program license agreement / license information, select the check box at the end of the page, and click **Next**.
2. for **Notices**, please read the notices, select the check box at the end of the page, and click **Next**.
3. for **Non IBM License**, please read the details on non-IBM license, select the check box at the end of the page, and click **Accept** to login in to your appliance.

> ⚠ In *License Agreement* page, from the drop-down available in the upper-right corner, you may choose the language from the list. For example, *English*.

If Single Sign-On is Disabled



To continue, please go to the step below to enter your Username and Password.

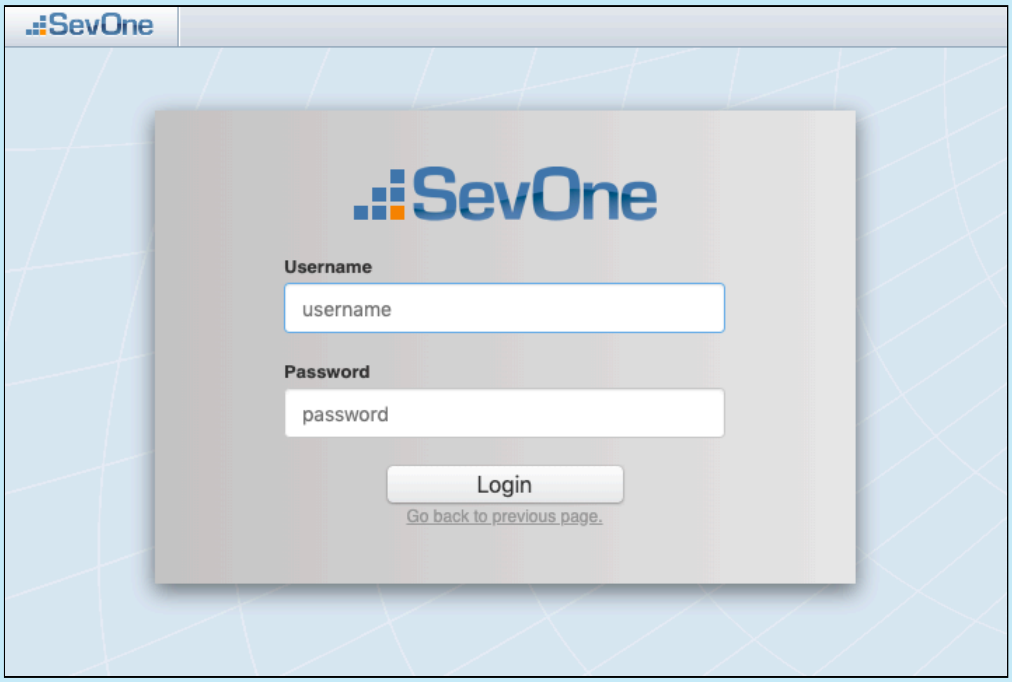
If Single Sign-On is Enabled

In this example, besides having the ability to login with **SevOne auth**, you see an Identity Provider, **Okta saml**, supported on the appliance.

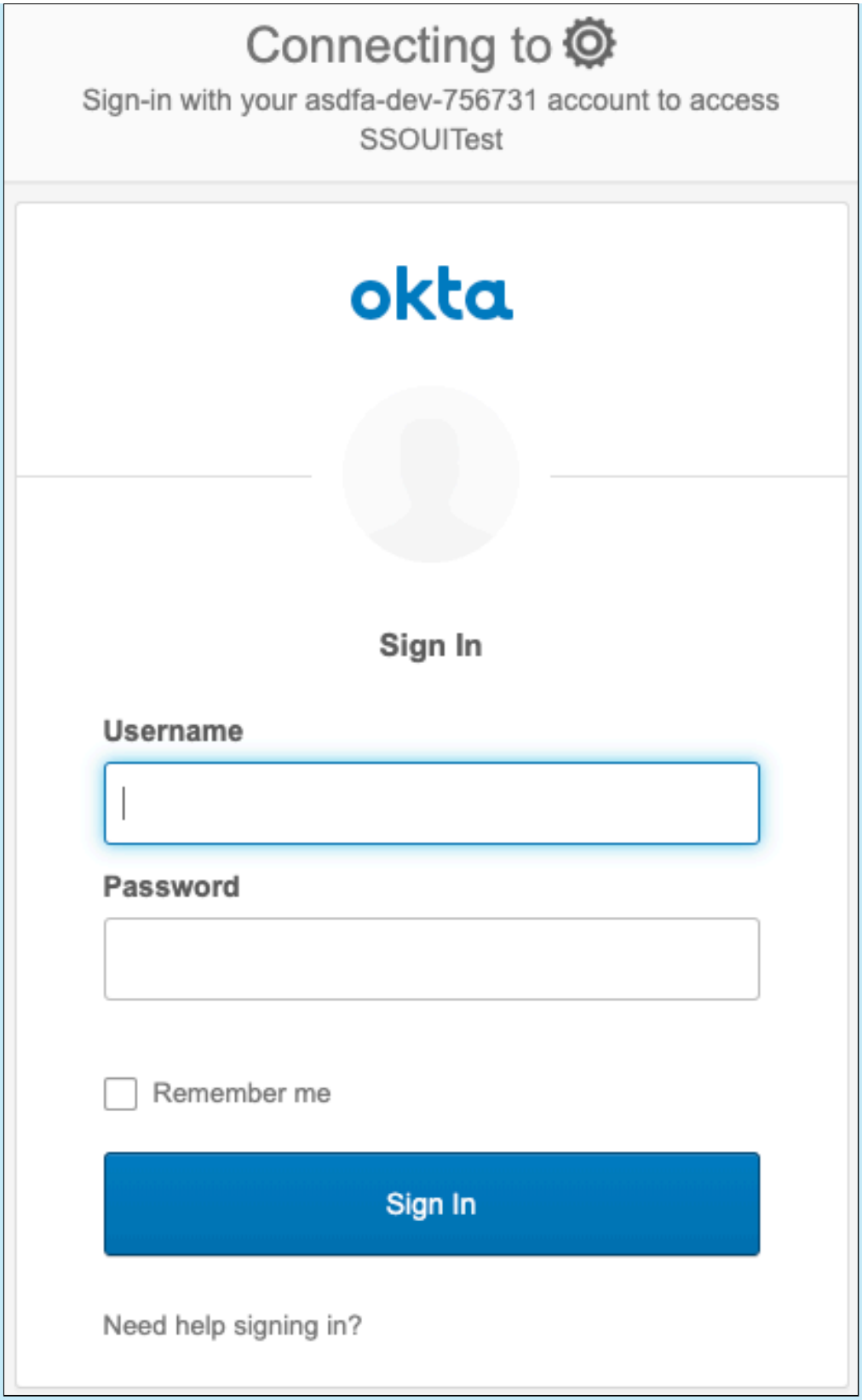
Okta saml is only an example. Besides **Log in with SevOne auth**, you will see your own list of Identity Providers.

Example

Click on **Log in with SevOne auth** or **Log in with Okta saml**.
If **Log in with SevOne auth** is selected, you will get the following display. Continue to the below to enter your Username and Password.



If **Log in with Okta saml** is selected, you will get the following display.



To continue, please refer to SevOne SAML Single Sign-On Setup Guide.

The Preferences page enables you to change your user information. **Cluster Manager** > **Cluster Settings** tab (see the **Security** subtab) enables you to define password security parameters such as minimum password length and password complexity. For details on **Preferences**, please refer to its section in *SevOne NMS User Guide*.

1. In the **Username** field, enter your SevOne NMS user name.
2. In the **Password** field, enter your password.
3. Click **Login** to log on to the application.

> For a new installation, the default system administrator account user name is **admin** and the password is **SevOne**. The admin user has a lot of power which, if abused, could cause considerable damage. After you log on for the first time, you will be prompted to change the default password.

3.1 User Time Settings

The first time you log on you can set your time zone. This setting is important to ensure that graphs, reports, and time spans display for the correct time.

1. Click the **Time Zone** drop-down and select a time zone. If a time zone does not appear in the list, the **Cluster Manager** > **Cluster Settings** tab > **Devices** subtab enables you to add time zones.
2. Select the **Never Ask Me Again** check box if you do not want to be prompted to update to the SevOne NMS time zone upon logon when the browser time zone is different. If you travel, you may want to leave this check box clear so your experience is always in your current time zone.
3. Click **Save**.

3.2 Change Password

If your user manager defines your account to force you to change your password when you log on for the first time, the Configuration pop-up includes fields to change your password.

1. In the **Old Password** field, enter your current password.
2. In the **New Password** field, enter your new password.
3. In the **Confirm Password** field, enter your new password a second time.
4. Click **Change Password**.

3.3 Default Passwords

It is highly recommended that default passwords are changed.

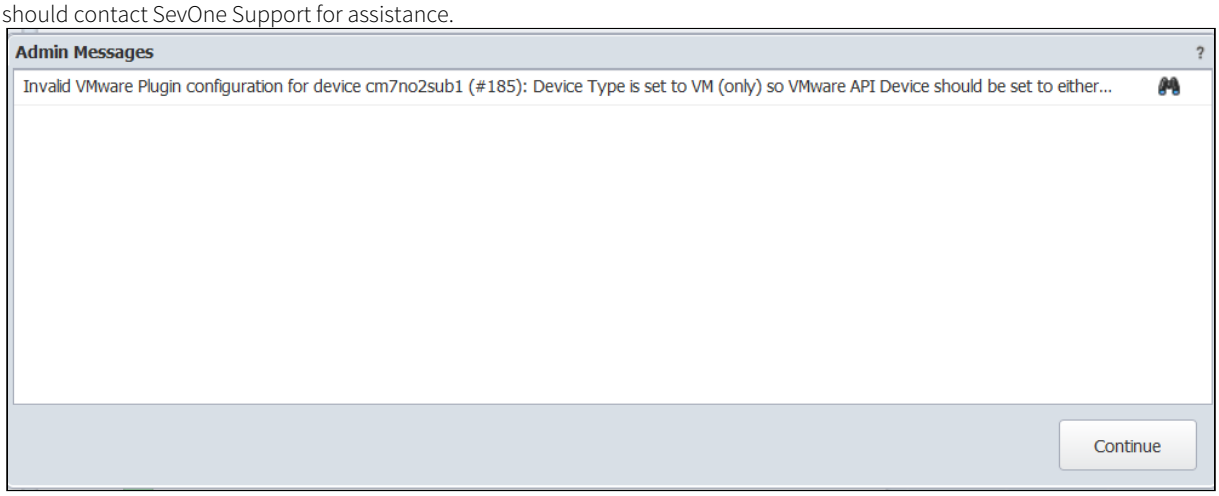
If the default password for any of the users is not changed, in the **Admin Messages** pop-up, it lists the users that require the password to be changed.

- **GUI users**: admin, SevOneStats
- **System Users**: root, admin, support

4 Administrative Messages

An Admin Messages page appears when users with administration roles log on and one or more of the following conditions exist.

- "Peer <peer name> is at <n> capacity." This message indicates that a peer in your cluster exceeds its license capacity. A peer does not discover any new devices or poll additional objects when a peer reaches its license capacity. See the Cluster Manager help topic for how to manage the elements each peer monitors.
- "Peer <peer name> dropped <n> flow records ... from IP <n> ..." This message indicates that a flow interface has sent a flow that exceed the Max Flow Duration you enter on the Cluster Manager > **Cluster Settings** tab. This is usually due to an improperly configured router which results in the router sending inaccurate flow data.
- "Your kernel version does not support some user action logging." This message indicates that some user action logging cannot be performed because your software uses a kernel that is less than 2.6.36. To find your kernel version number, click the **PHP Statistics** link on the About page.
- You can choose to display an administrative message when SevOne NMS software updates are available. You trigger the ability to display this message from the Cluster Manager > **Updates** tab.
- "Neither appliance in your Hot Standby Appliance peer pair with IP addresses <n> and <n> is in an active state." This message indicates that neither appliance in a Hot Standby Appliance (HSA) peer pair is actively polling data from your network. The Cluster Manager appliance level  enables you to correct this situation.
- "Both appliances in your Hot Standby Appliance peer pair with IP addresses <n> and <n> are either active or both appliances are passive." This message indicates that both appliances in a Hot Standby Appliance peer pair are attempting to perform the same role. Both appliances in the pair can end up in an active state when the Internet connection between the appliances is interrupted. The Cluster Manager appliance level  enables you to correct this situation.
- "SevOne NMS cannot determine the status of one of the appliances in your Hot Standby Appliance peer pair with IP addresses <n> and <n>. Please contact SevOne Support." If the peer is not turned off or disconnected from your network, you should contact SevOne Support for assistance.



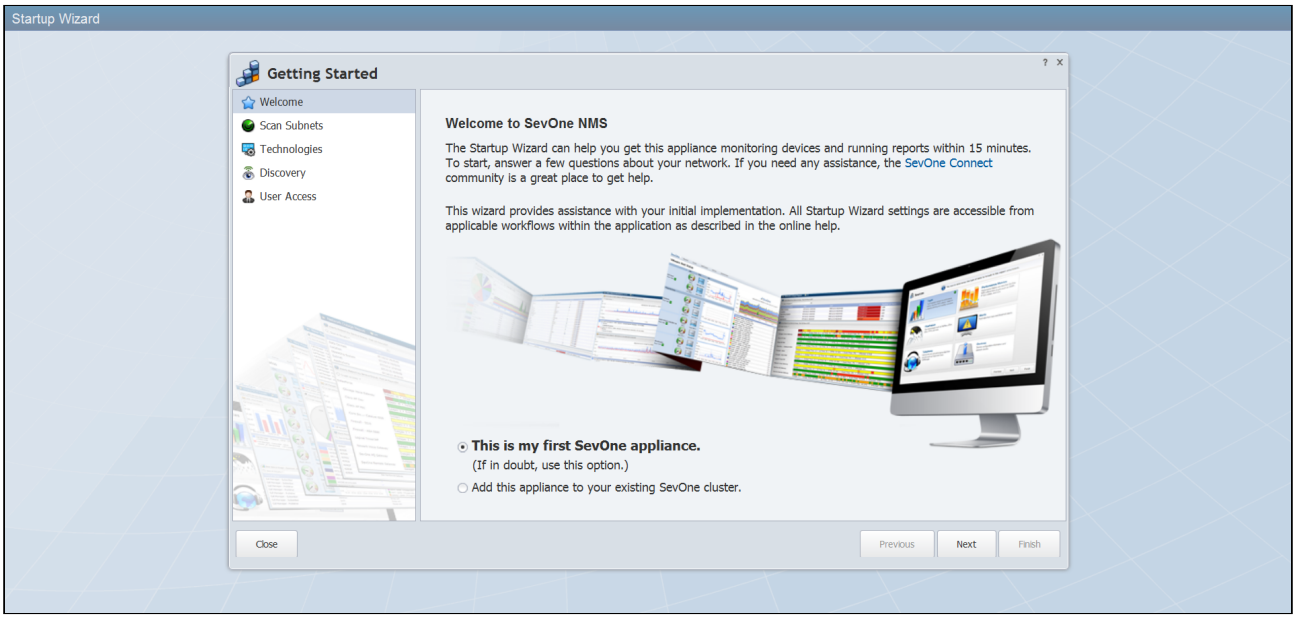
Make note of the messages and click **Continue**.

5 Startup Wizard

5.1 Initial Implementation for Administration Role Members

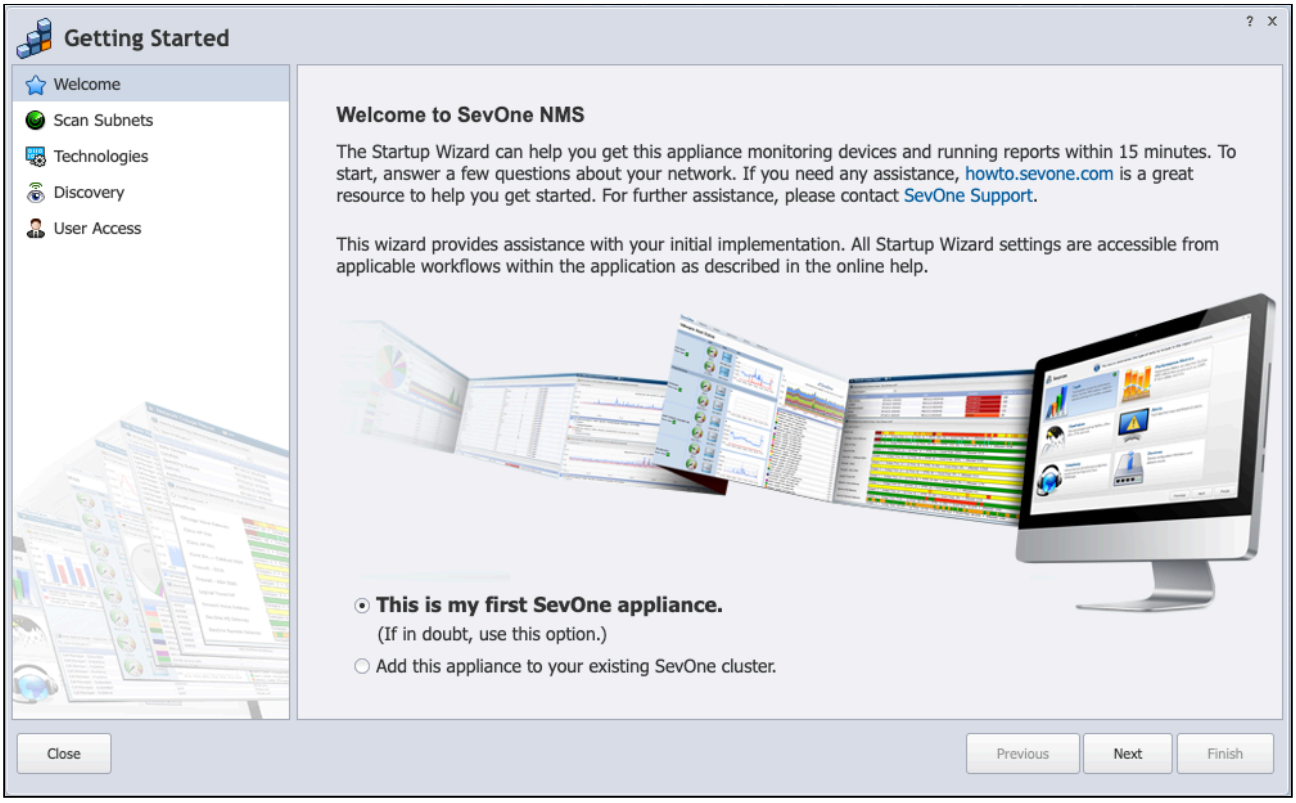
When the admin user logs on for the first time, the Startup Wizard appears to guide you through the initial steps to get devices you want to monitor and poll into SevOne NMS for discovery. If this is your first SevOne appliance, proceed through the Startup Wizard.

Users with administration roles can access the Startup Wizard from the navigation bar. However, the intent of the Startup Wizard is to help with your initial SevOne NMS implementation. All Startup Wizard workflows are duplicated within the application as noted in this document. Click the Administration menu and select Startup Wizard.



5.2 Startup Wizard - Welcome

The Welcome page on the Startup Wizard provides a link to howto.sevone.com which is a great source to help you get started.



5.2.1 First Appliance Option

To start a new SevOne NMS implementation, select the **This is my first SevOne appliance** option and click **Next** to display the Scan Subnets page on the Startup Wizard.

5.2.2 Existing Cluster Option

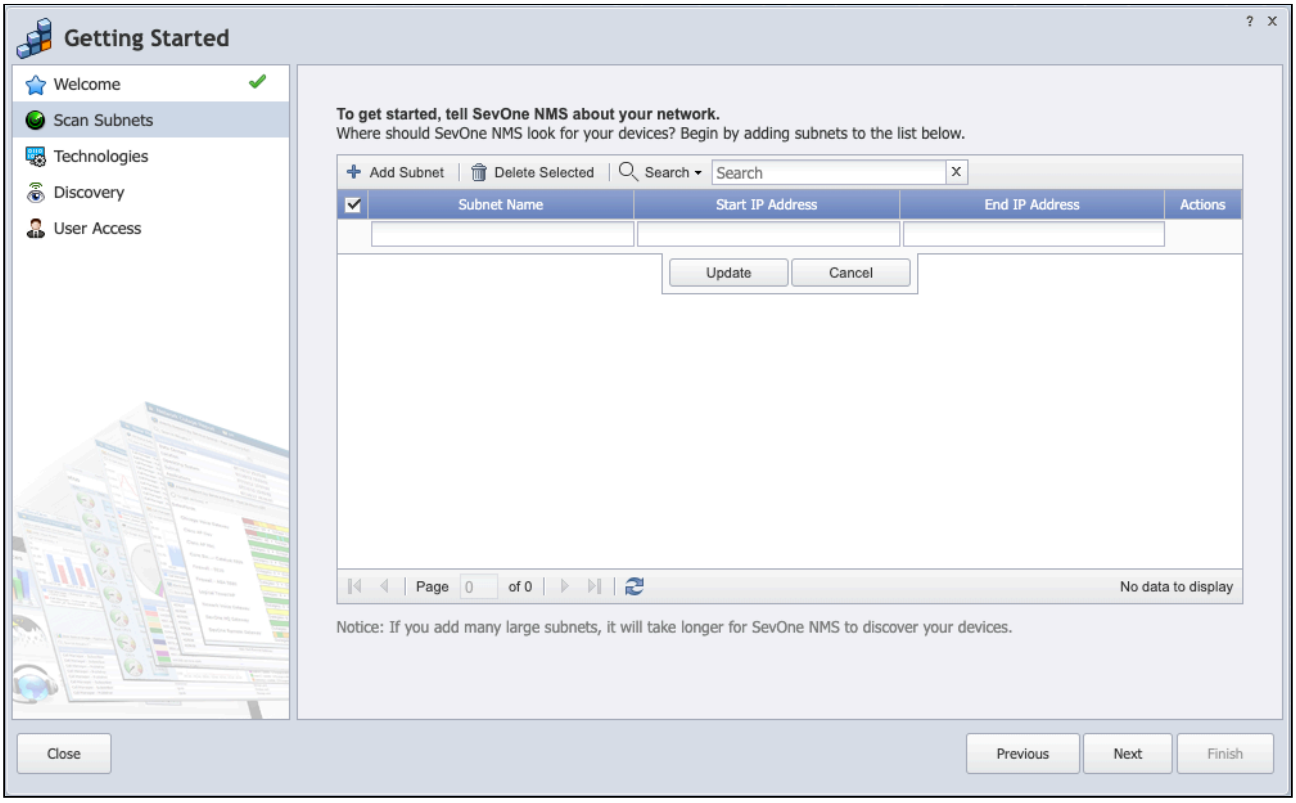
> ⚠ If you intend to add this appliance as a new peer to an existing cluster, **DO NOT ADD DEVICES TO THIS APPLIANCE** until after you add this appliance to your cluster. There is no way to combine the device databases of non-clustered/non-peered appliances.

1. Select the **Add this appliance to your existing SevOne cluster** option and click **Next** to access the Cluster Manager Integration tab where you can add the appliance as a new peer to your cluster.
2. Use the Device Mover to move devices to the new peer after you add the appliance to the cluster.

> ⓘ If this is a new Hot Standby Appliance, skip the Startup Wizard and contact SevOne Support to add the appliance to your cluster.

5.3 Startup Wizard - Scan Subnets

The Scan Subnets wizard page enables you to create IP address ranges to scan your network for items that can be pinged.

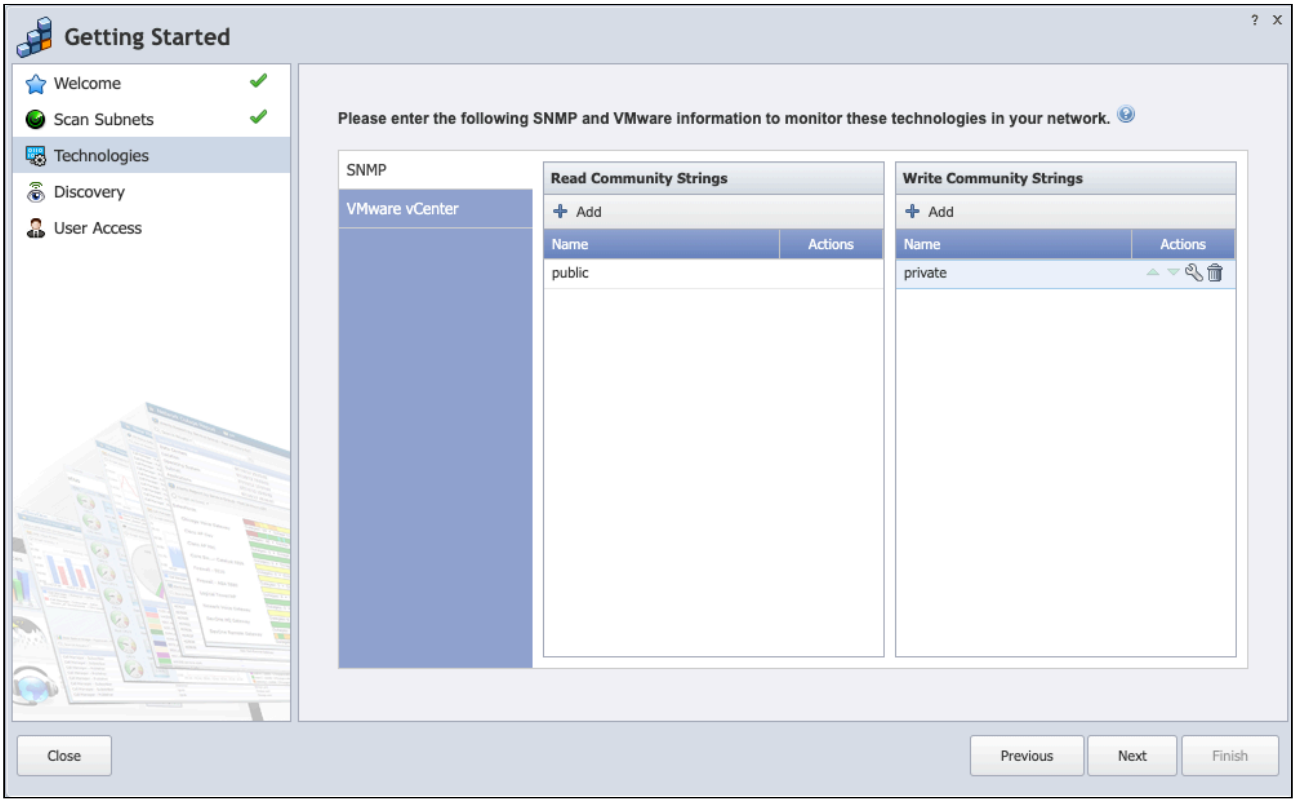


Each IP address range assigns devices to the peer that you are logged onto, creates devices, checks SNMP, and groups devices into a device group with the name you give to the subnet. All subnets are scanned one time when you click the Finish button. The entries you make on this wizard page are duplicated on the **Discovery Manager** > **Watched Subnets** tab. The **Device Manager** displays the devices that data is polled from. Please refer to sections **Discovery Manager** and **Device Manager** in *SevOne NMS User Guide* for details.

1. Click **Add Subnet** to add a row to the list.
2. In the **Subnet Name** field, enter the name of the subnet.
3. In the **Start IP Address** field, enter the low end of the IP address range.
4. In the **End IP Address** field, enter the high end of the IP address range.
5. Click **Update** to add the subnet to the list.
6. Repeat the previous steps to create additional subnets.
7. Click **Next**.

5.4 Startup Wizard - Technologies Page

When you enable devices to send data to SevOne NMS, default settings enable you to monitor many technologies on the devices you add to SevOne NMS. SNMP and VMware require you to enter some information about your network and the Technologies page enables you to get started.



5.4.1 SNMP

The SNMP subtab enables you to enter the community strings SevOne NMS needs to monitor SNMP data. You can update these settings on the Cluster Manager > **Cluster Settings** tab > **SNMP** subtab.

1. In either the **Read Community Strings** column or the **Write Community Strings** column click **Add** to add a row to the list.
2. In the **Name** field, enter the community string.
3. Click **Update** to add the string.
4. Repeat the previous steps to add additional strings.
5. Click or to move the string up or down through the list. SevOne NMS tries each string in the sequence in which they appear and stops at the first successful string.

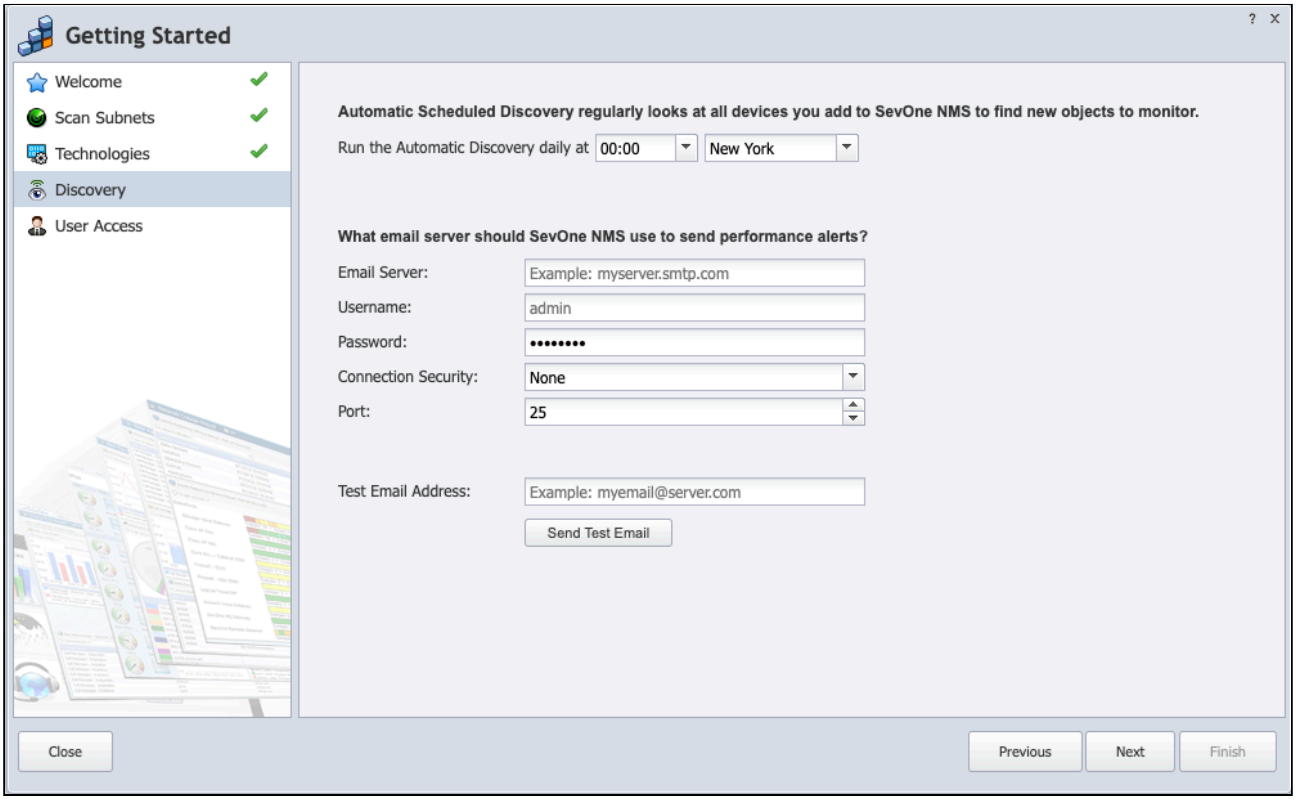
5.4.2 VMware vCenter

The VMware vCenter subtab enables you to enter the VMware vCenter login credentials. This enables SevOne NMS to discover and monitor the virtual hosts and virtual machines on the vCenter. The **VMware Browser** enables you to add additional vCenters and to manage the devices from which to poll VMware data. Please refer to section **VMware Browser** in *SevOne NMS User Guide* for details.

1. In the **VMware vCenter IP Address** field, enter the IP address of the VMware vCenter.
2. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the vCenter.
3. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the vCenter.
4. Select the **Automatically Discover Devices** check box to poll the vCenter's hosts and virtual machines on a daily basis. Leave clear to limit the amount of data from the vCenter's hosts and virtual machines.
5. Click **Next**.

5.5 Startup Wizard - Discovery

The Discovery wizard page enables you to set the time for SevOne NMS to perform the daily Automatic Discovery process and to define the email server that SevOne NMS uses to email reports and alerts.



5.5.1 Discovery

Discovery is the process to query and update information about the devices that are in SevOne NMS. Device discovery creates new objects in SevOne NMS, updates existing objects, and ultimately deactivates and deletes unused objects.

- **Manual Discovery** - The Manual Discovery process runs every two minutes to scan the devices in SevOne NMS that you mark for discovery.
- **Automatic Discovery** - The Automatic Discovery process tests the various plugins/technologies you configure for each device and updates the device's current state.

You define the Automatic Discovery time for each peer in the cluster on the Cluster Manager > **Peer Settings** tab.

1. Click the **Run the Automatic Discovery daily at** drop-down and select the time to run the Automatic Discovery process.
2. Click the second drop-down and select the time zone.

5.5.2 Email Server

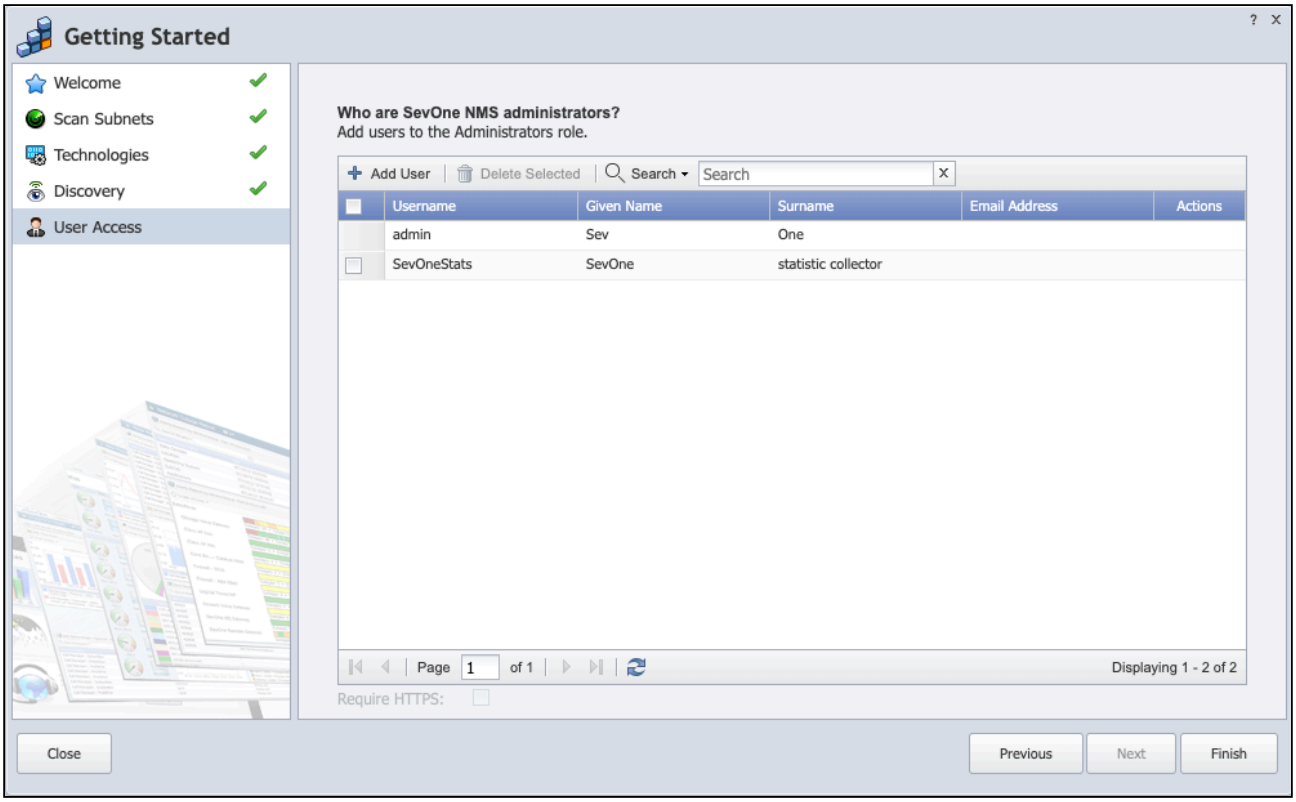
The rest of the page enables you to enter the email server information for SevOne NMS to use to notify users of alerts and to send reports to users. You can update these settings on the Cluster Manager > **Cluster Settings** tab > **Email** subtab.

> The email server must be able to accept large attachments because a .pdf report can be over 20MB. Please refer to section **Email Tips and Tricks** in *SevOne NMS User Guide* for additional details on email setup.

1. In the **Email Server** field, enter the hostname or IP address of the SMTP email server for SevOne NMS to use to send emails.
2. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the email server.
3. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the email server.
4. Click the **Connection Security** drop-down and select a connection security protocol.
5. In the **Port** field, enter the port on the email server for SevOne NMS to use.
6. In the **Test Email Address** field, enter the email address to which you want to send a test email.
7. Click **Send Test Email** to send a test email to the address you enter in the previous step.
8. Click **Next**.

5.6 Startup Wizard - User Access

The User Access page enables you to add users who are to be members of the Administrators user role.

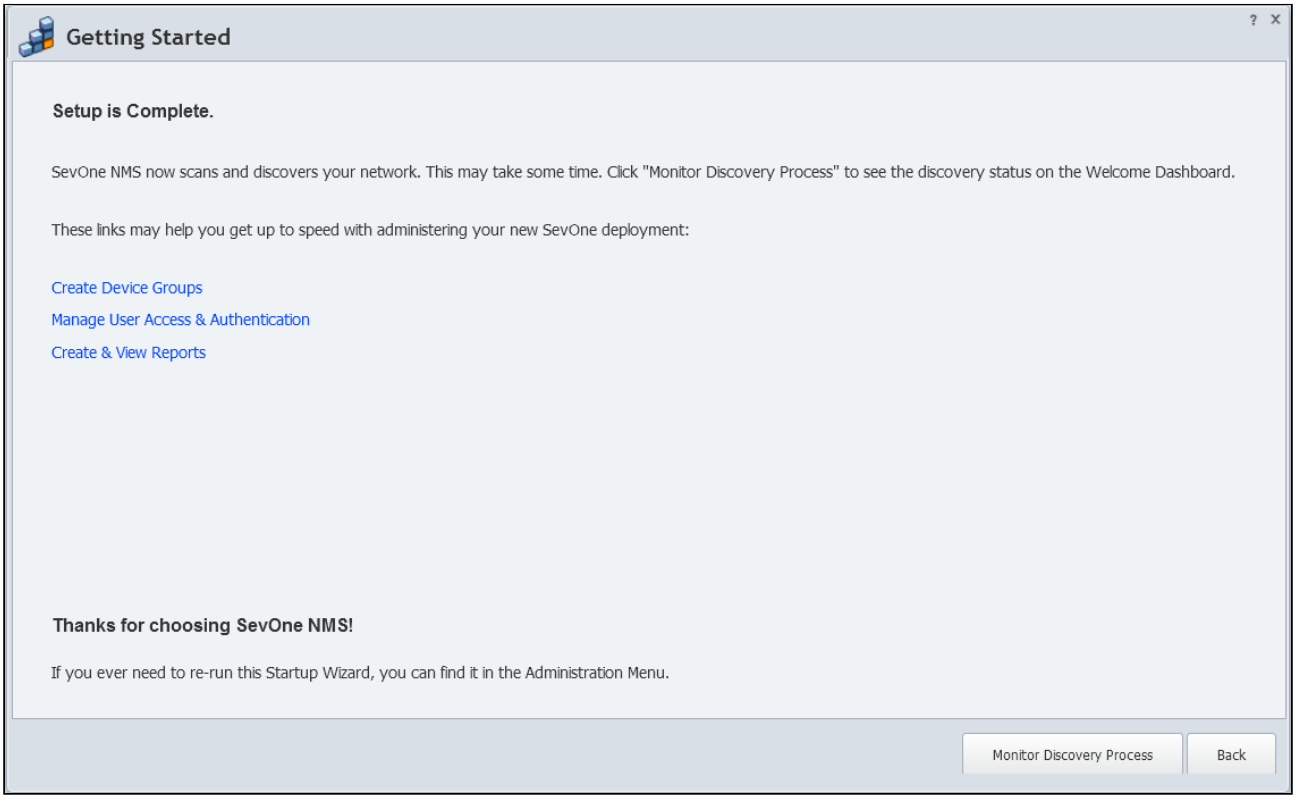


You must define the email server before you can add Administrators role users. You add all other users with any user role on the User Role Manager. Each user can update their user settings, except username, on the **Preferences** page. Please refer to section **Preferences** in *SevOne NMS User Guide* for details.

1. Click **Add User** to add a row to the list.
2. In the **Username** field, enter the name for the user to enter into the Username field on the Login page. After you save the user information, you cannot edit the Username.
3. In the **Given Name** field, enter the given name to display.
4. In the **Surname** field, enter the surname to display.
5. In the **Email Address** field, enter the email address where you want SevOne NMS to send emails to the user.
6. Click **Update** to save the user credentials. SevOne NMS sends an email to the user with the user's log on credentials.
7. Click **Finish** to start the scan of any subnets you define on the Scan Subnets wizard page and to display the Setup is Complete wizard page.

5.7 Startup Wizard - Setup is Complete

The Setup is Complete wizard page provides links to workflows that help get you started.



The Setup is Complete wizard page provides links to help you get started.

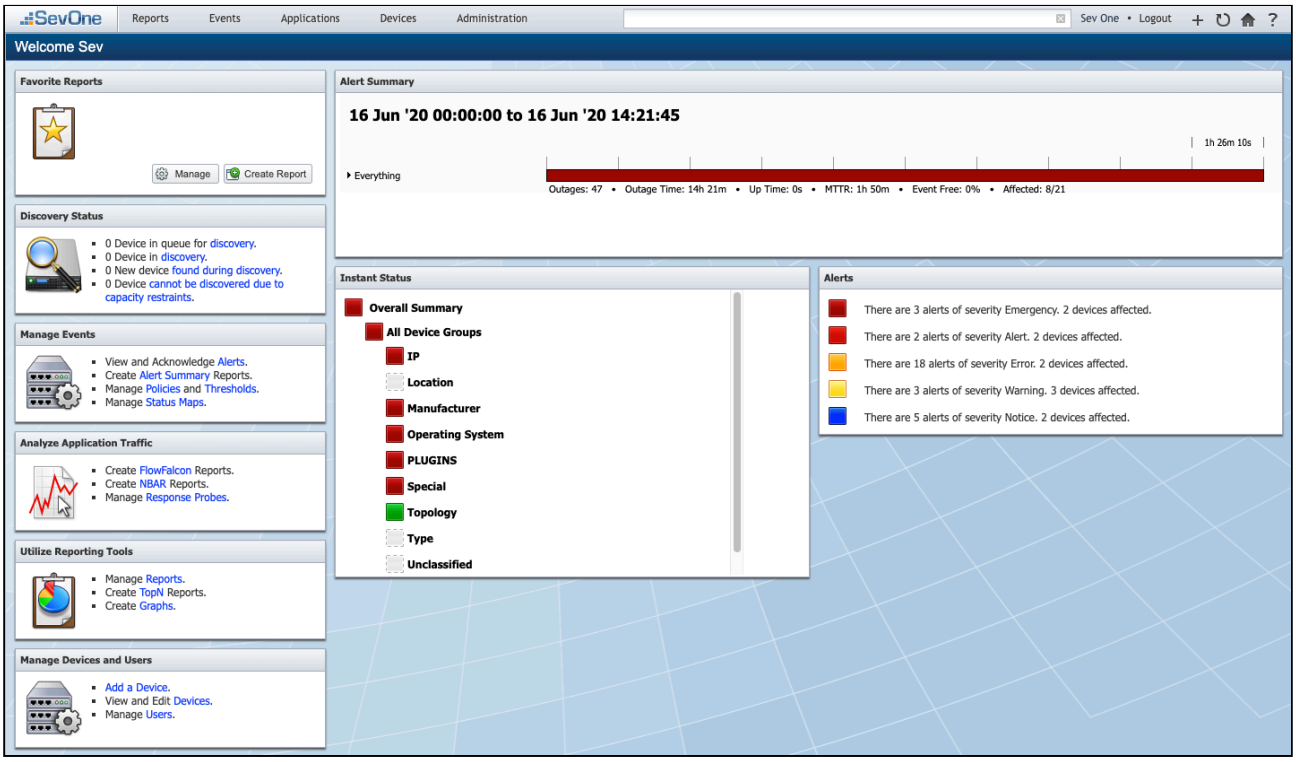
- Click **Create Device Groups** to navigate to the **Device Groups** page where you segment the devices in your network for user access, reports, and alerts. Please refer to section **Device Groups** in *SevOne NMS User Guide* for details.
- Click **Manage User Access & Authentication** to navigate to the User Manager page where you manage user information, credentials, and user role assignments.
- Click **Create & View Reports** to navigate to the **Report Manager** page that provides access to the workflows that enable you to combine and customize several graphs, tables, and other individual reports into a single easy to retrieve report. Please refer to section **Report Manager** in *SevOne NMS User Guide* for details.
- Click **Monitor Discovery Process** to navigate to the Welcome Dashboard.

6 Dashboard

The first page that appears when you log on is the Dashboard. The Dashboard is a gateway to the most common components in the application. The dashboard that appears when a new user logs on for the first time is the Welcome Dashboard. User roles enable you to restrict access to SevOne NMS workflows and to restrict access to devices. When your user role does not permit you to use a workflow, that workflow does not appear for you. Contact your SevOne NMS administrators to discuss your security settings.

> Topics mentioned here can be found in *SevOne NMS System Administration Guide* and/or *SevOne NMS User Guide*.

The **Report Manager** enables you to select a report to be your custom dashboard to define what appears for you when you log on.



6.1 Navigation Bar

The navigation bar appears across the top of most pages. The left side of the navigation bar provides the following menus.

Reports

- Create Report
- Report Manager
- Instant Graphs
- TopN Reports

Events

- Alerts
- Alert Summary
- **Archives** - Alert Archive, Logged Traps, and Unknown Traps
- **Configuration** - Policy Browser, Threshold Browser, Trap Destinations, Trap Destination Associations, Trap Event Editor, and Trap v3 Receiver.
- Instant Status
- Status Map Manager

Applications

- FlowFalcon Reports
- NBAR Reports
- Probe Manager

Devices

- Device Manager
- Object Manager
- Device Mover
- Discovery Manager
- Grouping - Device Groups and Object Groups
- SNMP Walk
- VMware Browser

Administration

- About
- Access Configuration - Authentication Settings, User Manager, User Role Manager, and Session Manager
- API Docs - Provides access to REST API documentation for Version 2 and Version 3.
- Baseline Manager
- Cluster Manager
- Flow Configuration - Flow Rules, Flow Interface Manager, FlowFalcon View Editor, MPLS Flow Mapping, Network Segment Manager, Object Mapping, Device Mapping, and Protocols and Services
- Maintenance Windows
- Metadata Schema
- Monitoring Configuration - Calculation Editor, Device Types, MIB Manager, Object Rules, Object Subtype Manager, Object Types, SNMP OID Browser, xStats Log Viewer, and xStats Source Manager
- My Preferences
- Startup Wizard
- Work Hours

The right side of the navigation bar provides the following controls:

Search - Click in the Search field and enter a minimum of three characters to search throughout the application. Wildcards are implied after the first three characters and special characters are not allowed. You can use shortcuts and keywords to enhance your search. Search results appear as a menu, and each result is a link to the appropriate page, report, etc. Click Advanced Search or Show All for advanced search options.

<user name> - Click the <user name> to display the Preferences page, where you can change your password, given name, surname, email address, date format, time settings, and language.

Logout - Click to log off.

+ - Click to open an additional instance of SevOne NMS on a new browser tab. Additional instances are based on your initial log on. Any additional instances close when you close the first SevOne NMS window.

↻ - Click to return settings to their default or last saved settings on the current page.

⌂ - Click to return to the Dashboard. You can also click on the SevOne logo to return to the Dashboard.

? - Click to display page specific online help.

6.1.1 Favorite Reports

The Favorite Reports section provides links to view your favorite reports. Report workflows enable you to define your favorite reports and to designate one report to appear as your custom dashboard (Report Interactions) instead of the default Welcome Dashboard. This enables you to cater reports to various audiences and to view vital data as soon as you log on.

- Click Manage to access the Report Manager, where you access all reports and report workflows.
- Click Create Report to access the Report Attachment Wizard, where you create new reports.
- Click on a favorite report link to display the report interactions on a new tab, where you can customize the report with real-time data.

6.1.2 Discovery Status

The Discovery Status section provides links to access additional discovery and device information.

- Click the Devices links to access the Discovery Manager, where you view discovery information.
- Click the Devices links to access the Device Manager, where you view device information.

6.1.3 Manage Events

The Manage Events section provides links to access additional event information.

- Click the **Alerts** link to access the Alerts page, where you view and acknowledge current active alerts.
- Click the **Alert Summary** link to access the Alert Summary page, where you view active alerts and archived alerts.
- Click the **Policies** link to access the Policy Browser, where you manage policies, which are thresholds that apply to an entire device group.
- Click the **Thresholds** link to access the Threshold Browser, where you manage the thresholds that trigger events such as alerts and traps for a specific device.
- Click the **Status Maps** link to access the Status Map Manager, where you manage status maps that can depict the physical or geographic location of devices and objects.

6.1.4 Analyze Application Traffic

The Analyze Application Traffic section provides links to access additional application traffic information.

- Click the **FlowFalcon** link to access the FlowFalcon Reports page, where you create flow technology reports.
- Click the **NBAR** link to access the NBAR Reports page, where you create Network-Based Application Recognition (NBAR) reports.
- Click the **Response Probes** link access the Probe Manager, where you manage Internet Protocol Service Level Agreement (IP SLA) data.

6.1.5 Utilize Reporting Tools

The Utilize Reporting Tools section provides links to access additional information about reporting tools.

- Click the **Reports** link to access the Report Manager, where you view and edit reports.
- Click the **TopN** link to access the TopN Reports page, where you create reports for projections and the top <n> devices.
- Click the **Graphs** link to access the Instant Graphs page, where you create performance metrics graphs.

6.1.6 Manage Devices and Users

The Manage Devices and Users section provides links to access device management and user management tools.

- Click the **Add a Device** link to access the New Device page, where you add a device to SevOne NMS.
- Click the **Devices** link to access the Device Manager, where you manage the devices SevOne NMS monitors.
- Click the **Users** link to access the **User Manager**, where you manage users.

6.1.7 Alert Summary

The Alert Summary section displays a summary of active alerts and archived alerts. Click anywhere in the Alert Summary to display the Alert Summary, where you can view additional alert details.

The **Cluster Manager** > **Cluster Settings** tab > **Login** subtab enables you to remove this section and the following sections from the Welcome Dashboard.

6.1.8 Instant Status

The Instant Status section displays the device group hierarchy color-coded to indicate the highest level active alert for the devices in each device group. Click anywhere in the Instant Status section to access the Instant Status page, where you can view additional active alert details.

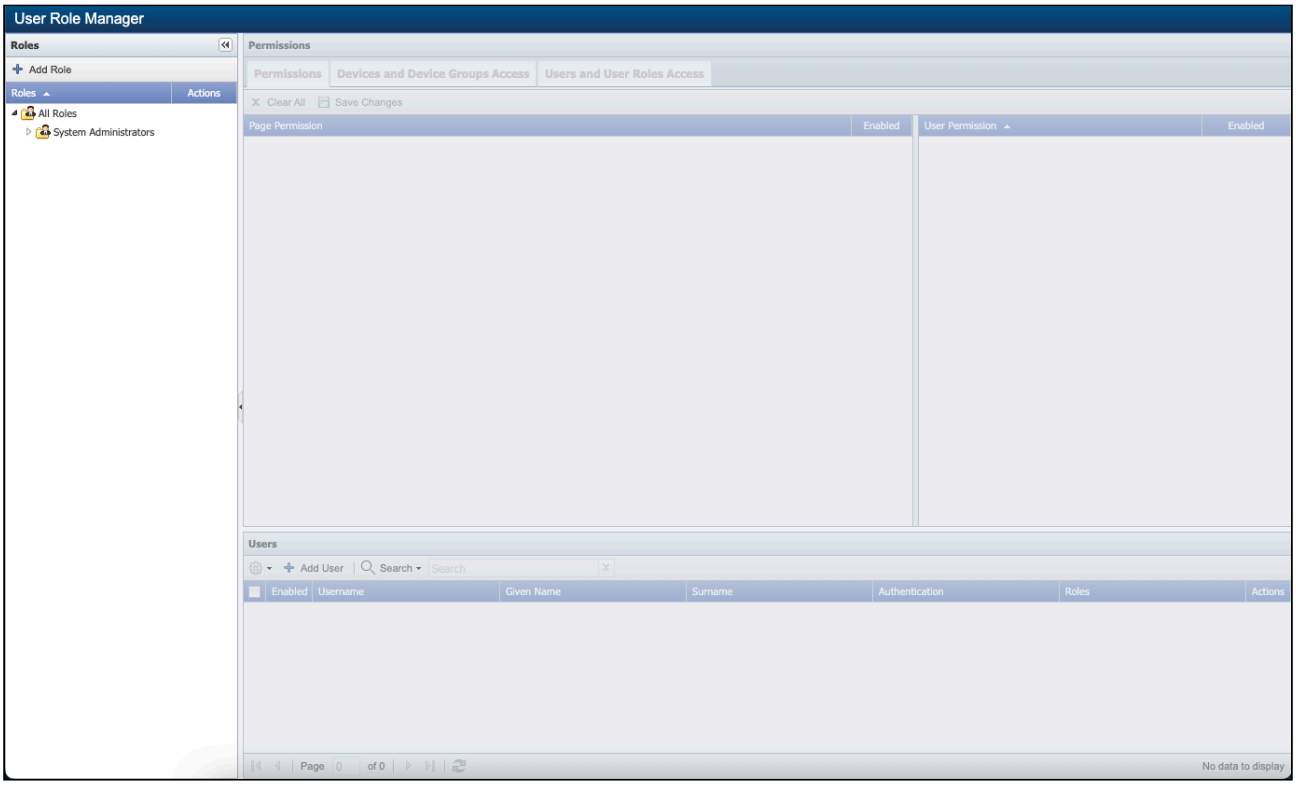
6.1.9 Alerts

The Alerts section displays the number of active alerts for each alert severity level. Click a colored square to access the Alerts page, where you can view additional alert details.

7 User Role Manager

The User Role Manager enables you to define the permissions, device/device group/device type access, and user/user role access that enables you to restrict what users can do and see in the application. User roles are hierarchical. Each lower level (child) user role can have either the same permissions as its parent user role or a subset of the permissions of its parent user role. You can assign users to multiple roles.

To access the User Role Manager from the navigation bar, click the **Administration** menu, select **Access Configuration**, and then select **User Role Manager**.



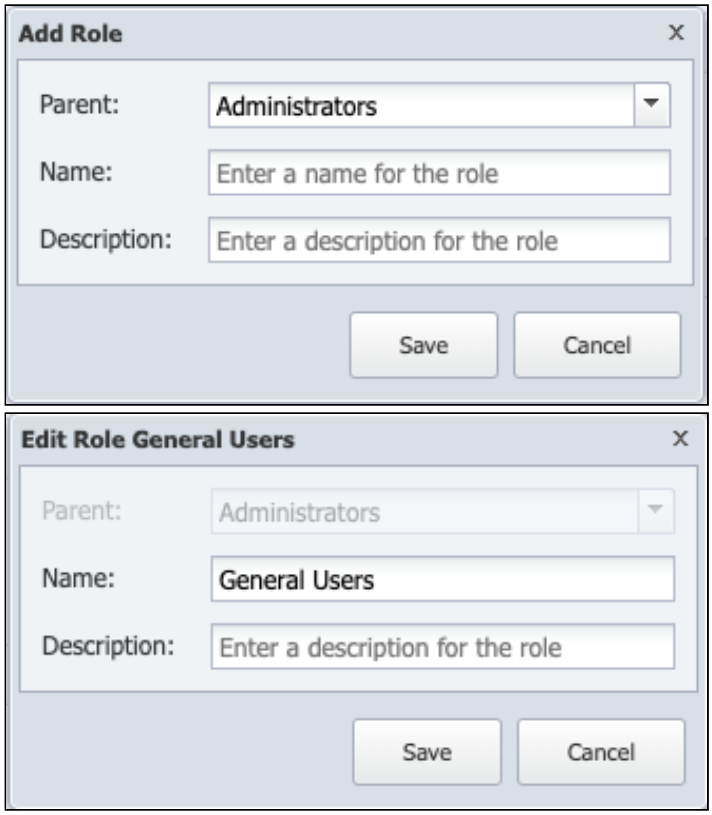
7.1 Roles

The user roles to which you are granted Role View permission appear in the role hierarchy on the left side. The Users and User Roles Access tab enables you to define roles that have access to view roles but not edit roles that are above their role in the hierarchy.

1. Click **Add Role** or  to display the Add/Edit Role pop-up.

> To edit an existing user role, select the role from the role hierarchy and click  to display the **Edit Role** pop-up. You can edit the user role name and description. While editing a role, parent cannot be changed.

Example: Add / Edit User Role



2. For a new role, click the **Parent** drop-down and select the role under which to add the role in the role hierarchy. You cannot edit this field after you click Save.
3. In the **Name** field, enter the name of the role.
4. In the **Description** field, enter a description for the role.
5. Click **Save**.

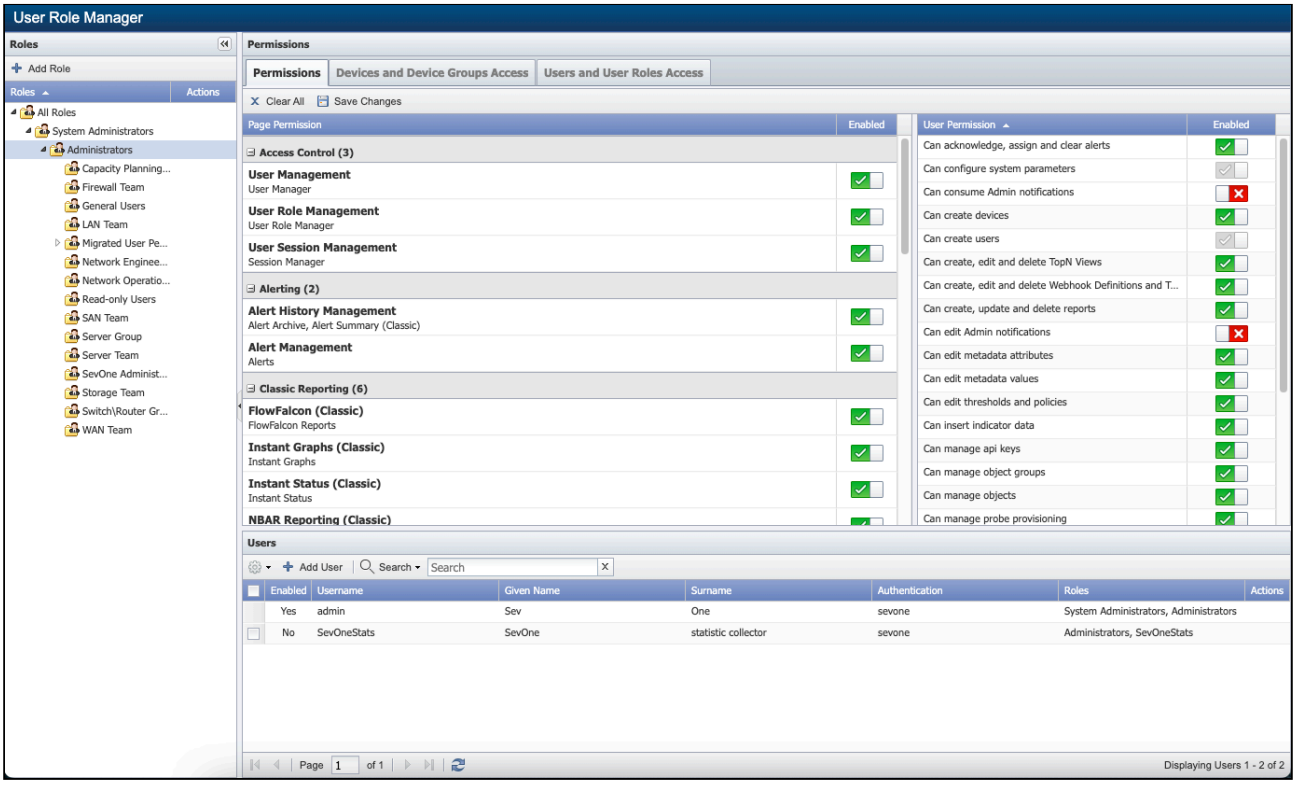
LDAP groups are associated with SevOne User Roles nested in the LDAP folder. The LDAP sync process will automatically perform the following actions:

- Create or delete User Roles within the LDAP folder hierarchy for any LDAP groups present during the sync.
- Create new user accounts for any users present in the LDAP groups.
- Add or remove User Roles to individual user accounts based on their LDAP group assignment.

LDAP roles created by the sync will have no permissions by default and must be maintained manually. If LDAP group assignment is changed for a user, the next LDAP sync will modify the user's roles in the NMS accordingly.
User roles not nested within the LDAP roles folder can be assigned to LDAP users but require manual management by an administrator.

7.2 Permissions

The **Permissions** section provides three tabs (**Permissions**, **Devices and Device Groups Access**, and **Users and User Roles Access**) to enable you to define the permissions for each role. All permissions are cumulative and each tab provides a subset of the permissions a user needs to perform tasks.



Example

To enable users to acknowledge, assign, and clear alerts, the user role must have:

- Permissions
  - Page Permission - alert management.
  - User Permission - can Acknowledge, Assign, and Clear Alerts, and Can View Alerts.
- Devices and Device Groups Access - enable access to device groups/device types that contain the devices from which the user is to be able to manage alerts.

  If a Device Group is moved from one parent to another, the permissions for that device group are changed to inherit the permissions from the new parent.

- Users and User Roles Access - enable access to user roles that contain the users to which the user is to assign alerts.

As you enable permissions for higher level user roles, the same permission becomes available for the subordinate user roles. Conversely, when you disable permissions, the corresponding permissions are no longer available for the subordinate user roles.

The following buttons appear below the Permissions tabs.

- Click Clear All to reset the permissions on all tabs to their last saved settings.
- Click Save Changes to save the changes made to all permissions on all tabs.

7.2.1 Page Permission and User Permission

The Permissions tab enables you to define which pages in SevOne NMS users assigned to the role can access and what the user can do on those pages. Additional Devices and Device Groups permissions and/or Users and User Roles permissions are required to actually see device data on the page. In other words, access to pages does not necessarily enable users to see or do anything on that page.

7.2.1.1 Page Permission

The Page Permission section enables you to grant users access to view applicable pages. As you enable page permissions, the corresponding minimum user permissions are enabled. You can enable additional user permissions when you feel they are applicable but you cannot disable the minimal user permission the page requires.

> **Example**
>
> You can grant a role the user permission Can Acknowledge, Assign, and Clear Alerts but if you do not enable the page permission to the Alerts page, the users in the role cannot see the workflows in the application that provide the ability to acknowledge, assign, or clear alerts.

Perform the following steps to manage the page permissions for a role.

1. In the Roles hierarchy, select a role to populate the Permissions tabs with the permissions for the role you select.
2. Select the **Permissions** tab, if needed.
3. In the Page Permission column, enable each permission to grant the users access to the pages listed.

When you disable all page permissions, an enabled user has permission to access the following pages.

- About
- Dashboard
- My Preferences (Please refer to section **Preferences** in *SevOne NMS User Guide* for details).

7.2.1.1.1 Access Control

- Enable **User Management** to grant access to: User Manager.
  - Corresponding User Permissions: **Can create users**
- Enable **User Role Management** to grant access to: User Role Manager.
  - Corresponding User Permissions: **Can create users**
- Enable **User Session Management** to grant access to: Session Manager.
  - Corresponding User Permissions: **None**

7.2.1.1.2 Alerting

- Enable **Alert History Management** to grant access to: Alert Archives and Alert Summary.
  - Corresponding User Permissions: **Can view alert history**, **Can view alerts, Can view reports**
- Enable **Alert Management** to grant access to: Alerts.
  - Corresponding User Permissions: **Can view alerts**, **Can view reports**

7.2.1.1.3 Classic Reporting

- Enable **FlowFalcon (classic)** to grant access to FlowFalcon Reports.
  - Corresponding User Permissions: **Can view flow data**, **Can view reports**
- Enable **Instant Graphs (classic)** to grant access to: Instant Graphs.
  - Corresponding User Permissions: **Can view reports**
- Enable **Instant Status (classic)** to grant access to: Instant Status.
  - Corresponding User Permissions: **Can view alerts**, **Can view reports**
- Enable **NBAR Reporting (classic)** to grant access to: NBAR Reports.
  - Corresponding User Permissions: **Can view reports**
- Enable **Status Maps (classic)** to grant access to: Status Map Manager.
  - Corresponding User Permissions: **Can view alerts**, **Can view reports**
- Enable **TopN (classic)** to grant access to: TopN Reports.
  - Corresponding User Permissions: **Can view reports**

7.2.1.1.4 Device Management

- Enable **Device Manager** to grant access to: Device Manager and SNMP Walk.
  - Corresponding User Permissions: **None**. You can enable the **Can create devices** user permission to grant users the ability to manage devices. The Devices and Device Groups Access tab enables you to limit which devices users can see.
- Enable **Discovery Management** to grant access to: Discovery Manager.

    - Corresponding User Permissions: **None**. You can enable the **Can create devices** user permission to grant users permission to manage devices. The Devices and Device Groups Access tab enables you to limit which devices users can see.
- Enable **Probe Provisioning** to grant access to: Probe Manager and the **Proxy Ping** configuration on the **Edit Device** page.
    - Corresponding User Permissions: **None**
- Enable **VMware Browser** to grant access to: VMware Browser.
    - Corresponding User Permissions: **None** - Limits set from Devices and Device Groups Access tab

7.2.1.1.5 Metadata

- Enable **Metadata Attributes** to grant access to: Metadata Schema.
    - Corresponding User Permissions: Requires **Can edit metadata attributes** to edit the metadata attributes this page permission enables you to view.
- Enable **Metadata Values** to grant access to the Edit Metadata workflow from the following pages: Device Types, Device Groups, Device Manager, Edit Device, Object Types, and Object Manager.
    - Corresponding User Permissions: Requires **Can edit metadata values** to edit the values this page permission enables you to view.

7.2.1.1.6 Other

- Enable **Device Group Manager** to grant access to: Device Groups.
    - Corresponding User Permissions: **None** - Limits set from Devices and Device Groups Access tab
- Enable **Object Manager** to grant access to: Object Manager.
    - Corresponding User Permissions: **None** - Limits set from Devices and Device Groups Access tab
- Enable **Report Manager** to grant access to: Report Attachment Wizard and Report Manager.
    - Corresponding User Permissions: **Can view reports**

7.2.1.1.7 System Administration

Enable **Cluster Configuration** to grant access to: Authentication Settings, Baseline Manager, Cluster Manager, Device Mover, and Work Hours.

    - Corresponding User Permissions: **Can configure system parameters**
- Enable **Flow Monitoring Configuration** to grant access to: Flow Interface Manager, Flow Rules, FlowFalcon View Editor, MPLS Flow Mapping, Network Segment Manager, Object Mapping, and Protocols and Services.
    - Corresponding User Permissions: **Can configure system parameters**
- Enable **Maintenance Window Configuration** to grant access to: Maintenance Window Manager.
    - Corresponding User Permissions: **Can configure system parameters**
- Enable **Object Group Manager** to grant access to: Object Groups.
    - Corresponding User Permissions: **Can configure system parameters**
- Enable **Polling Configuration** to grant access to: Calculation Editor, Device Types, MIB Manager, Object Rules, Object Subtype Manager, Object Types, and SNMP OID Browser.
    - Corresponding User Permissions: **Can configure system parameters**
- Enable **Threshold Configuration** to grant access to: Policy Browser and Threshold Browser.
    - Corresponding User Permissions: **Can edit thresholds and policies**, **Can view thresholds and policies**
- Enable **Trap Configuration** to grant access to: Logged Traps, Trap Destinations, Trap Destination Associations, Trap Event Editor, Unknown Traps, Trap v3 Receiver.
    - Corresponding User Permissions: **configure system parameters**
- Enable **Webhook Configuration** to grant access to: Webhook Definitions.
    - Corresponding User Permissions: **Can create, edit and delete Webhook Definitions and Templates**
- Enable **xStats Configuration** to grant access to: xStats Log Viewer, and xStats Source Manager.
    - Corresponding User Permissions: **configure system parameters**

7.2.1.2 User Permissions

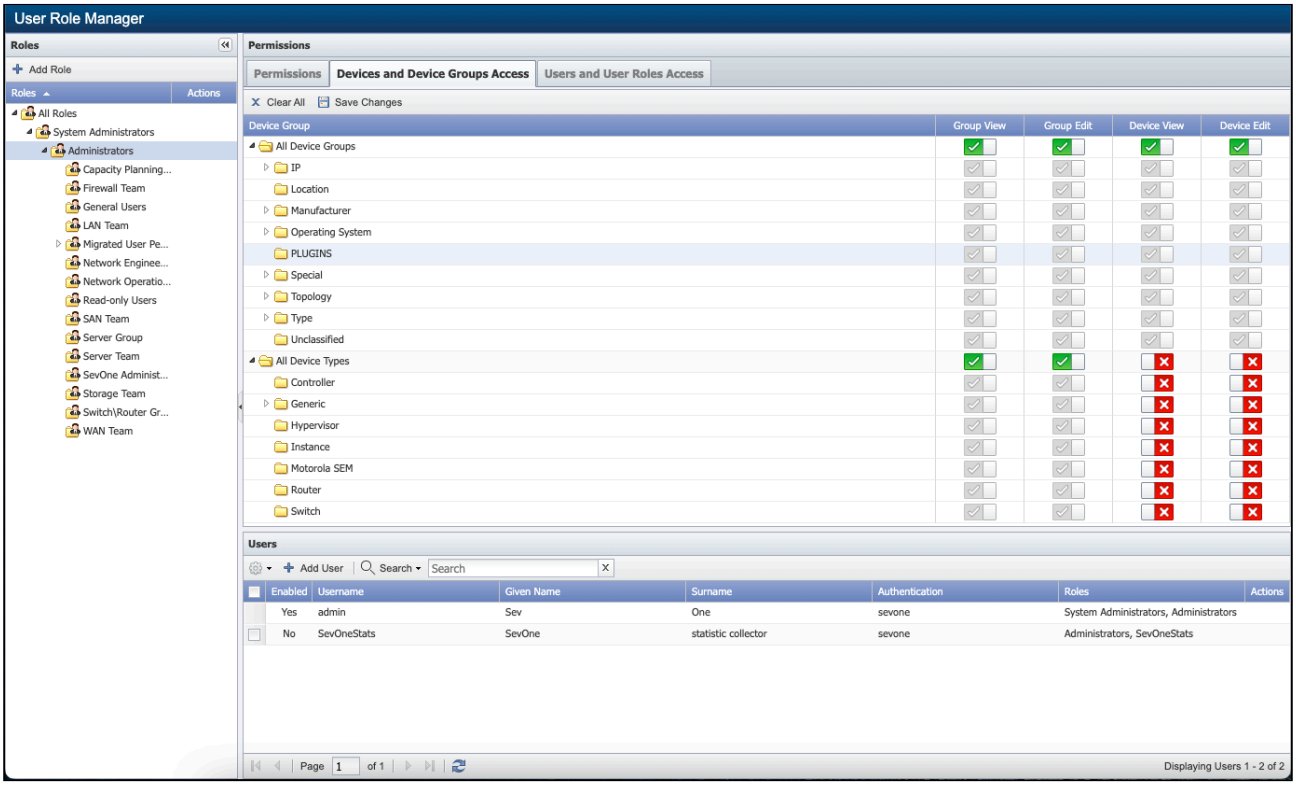
The User Permission section enables you to grant additional permissions to users in the role. User permissions are governed by the Page Permission settings and the user is further limited by their Devices and Device Groups Access and their Users and User Roles Access.

- Enable **Can acknowledge, assign, and clear alerts** to grant permission to acknowledge, assign, and clear alerts when you also enable appropriate Devices and Device Groups permissions and you enable page permission for Alert Management.
- Enable **Can configure system parameters** to grant permission to perform system administration tasks when you also enable appropriate Devices and Device Groups permissions and you enable appropriate page permissions.

- Enable **Can consume Admin notifications** to grant permission to a user to receive *prometheus alertmanager* email notifications.
- Enable **Can create devices** to grant permission to create, edit and delete device data when you also enable appropriate Devices and Device Groups permissions and you enable page permission for Device Manager.
- Enable **Can create users** to grant permission to create, edit, and delete user data when you also enable appropriate Users and User Roles permissions and you enable page permission for User Management and/or User Role Management.
- Enable **Can create, edit and delete TopN Views** to grant permission to create, edit and delete TopN Views based on role-based access control.
- Enable **Can create, edit and delete Webhook Definitions and Templates** to grant permission to create, edit and delete Webhook Definitions based on role-based access control.
- Enable **Can create, update and delete reports** to grant permission to create and save report data when you also enable appropriate Devices and Device Groups permissions and you enable page permission for Instant Graphs, Device Manager, NBAR Reports, Report Manager, and/or Status Map Manager. Users can create disposable reports from these pages and can detach individual reports to a comprehensive report that they can save to the Report Manager.
- Enable **Can edit Admin notifications** to grant permission to a user to enable/disable the delivery of *prometheus alertmanager* admin email notifications.
- Enable **Can edit metadata attributes** to grant permission to edit the metadata attributes on the Metadata Schema page.
- Enable **Can edit metadata values** to grant permission to edit the values for the metadata attributes that are specific to a device type, device group, device, object group, or object.
- Enable **Can edit thresholds and policies** to grant permission to edit the values for thresholds and policies.
- Enable **Can insert indicator data** to grant permission to insert indicator data. This is used in conjunction with the API.
- Enable **Can manage api keys** to grant permission to manage API keys. This is used in conjunction with the API.
- Enable **Can manage object groups** to grant permission to manage object groups.
- Enable **Can manage objects** to grant permission to manage objects.
- Enable **Can manage probe provisioning** to grant permission to provision probes via the Probe Manager when you enable appropriate Devices and Device Groups and you enable the Probe Provisioning page permission.
- Enable **Can perform discovery related tasks without permission checks** to grant permission to perform discovery tasks without the need for permission checks. This is used in conjunction with the API.
- Enable **Can receive alert notifications** to grant permission to receive email notifications from applicable traps, policies, and thresholds. This permission does not grant access to any workflows in SevOne NMS.
- Enable **Can view alert history** to grant permission to view archived alerts when you also enable appropriate Devices and Device Groups permissions and you enable page permission for Alert History Management.
- Enable **Can view alerts** to grant permission to view alerts when you also enable appropriate Devices and Device Groups permission and you enable the page permission for Alert Management.
- Enable **Can view flow data** to grant permission to view flow data in FlowFalcon reports when you also enable the appropriate Devices and Device Groups permission and you enable the page permission for FlowFalcon Reports.
- Enable **Can view reports** to grant permission to view report data when you also enable appropriate Devices and Device Groups permissions and you enable page permission for Instant Graphs, Device Manager, NBAR Reports, Report Manager, and/or Status Map Manager. Users assigned to the role can create disposable reports from these pages and can detach individual reports to a comprehensive report but they cannot save reports to the Report Manager.
- Enable **Can view thresholds and policies** to grant permission to view the values for thresholds and policies.
- Enable **Can view unmapped flow devices** to allow users to access flow devices that are not mapped to any other plugin device. In order to give the user access to a flow device that is mapped to another plugin device, user must have access to the plugin device.

7.2.2 Devices and Device Groups Access

The Devices and Device Groups Access tab enables you to grant permissions to view and/or edit specific device groups/device types and/or to view and/or edit specific devices.



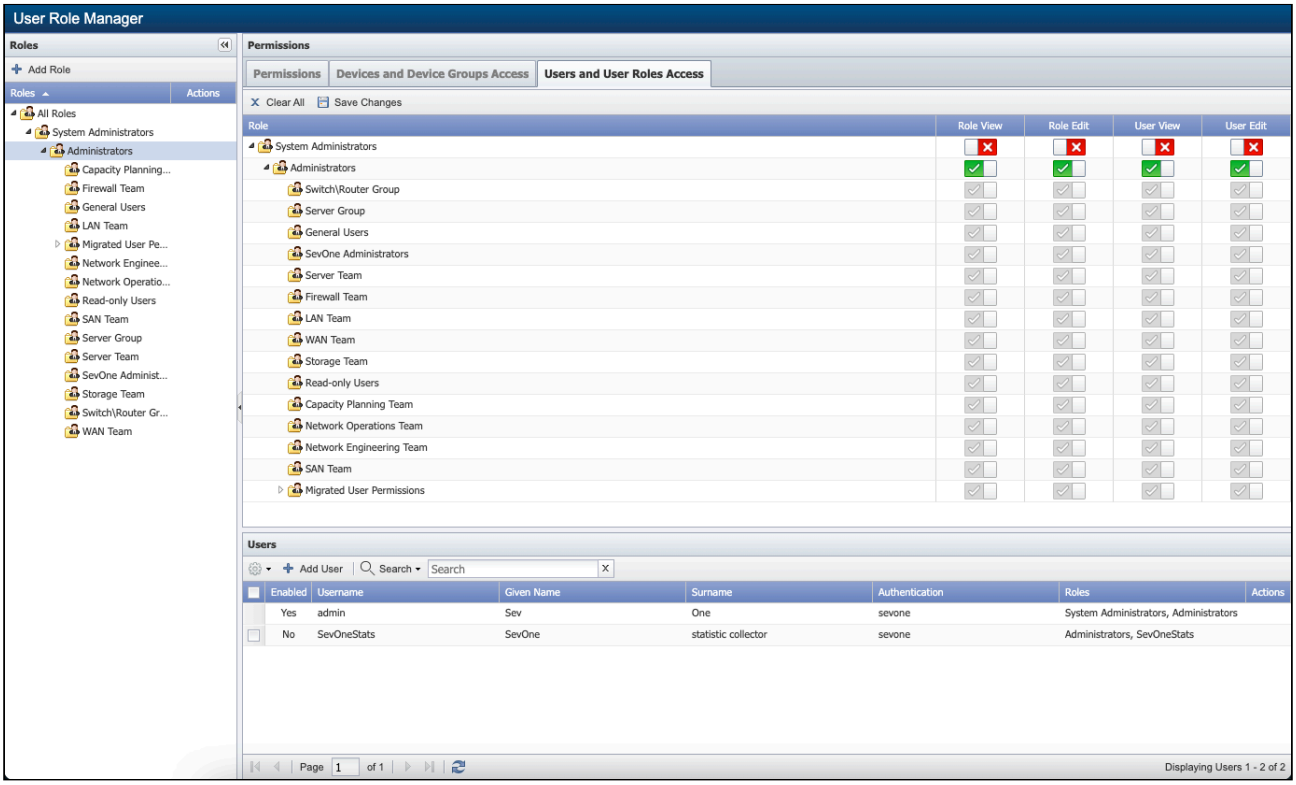
1. Select the **Devices and Device Groups Access** tab.
2. Enable the following to grant permission to view and/or edit device groups/device types and the devices within device groups/device types.

> As you enable the Devices and Device Groups permissions for higher level user roles, the same Devices and Device Groups permissions becomes available for the subordinate user roles. Conversely, when you disable the Devices and Device Groups permissions, the corresponding Devices and Device Groups permissions are no longer available for the subordinate user roles.

   a. Enable **Group View** to grant permission to view the name of the device group/device type. Users cannot see the devices that are members of the device group/device type or any corresponding device data until you enable Device View permission.
   b. Enable **Group Edit** to grant permission to edit the device group/device type name. Enable Device View permissions to grant permission to edit the list of devices that are members of the device group/device type.
   c. Enable **Device View** to grant permission to see the devices that are members of the device group/device type and applicable corresponding device data. If you disable the Group View permission, users can see the devices that are members of the device group/device type but cannot see the device group/device type name. When a device is mapped to a flow a device, this option allows users to see the flow device. Users have permissions to all device properties such as collected data, triggered alerts, etc. when they have access to the device. Permissions can be extended to flow devices as well via object mappings.
   d. Enable **Device Edit** to grant permission to edit the configuration of the devices that are members of the device group/device type.

7.2.3 Users and User Roles Access

The Users and User Roles Access tab enables you to define which user roles users can see and/or edit and which users the users can see and/or edit.



1. Select the **Users and User Roles Access** tab.
2. Enable the following permissions to grant permission to view and/or edit user roles and the users assigned to the user roles.

> As you enable the Users and User Roles permissions for higher level user roles, the same Users and User Roles permissions become available for the subordinate user roles. Conversely, when you disable the Users and User Roles permissions, the corresponding Users and User Roles permissions are no longer available for the subordinate user roles.

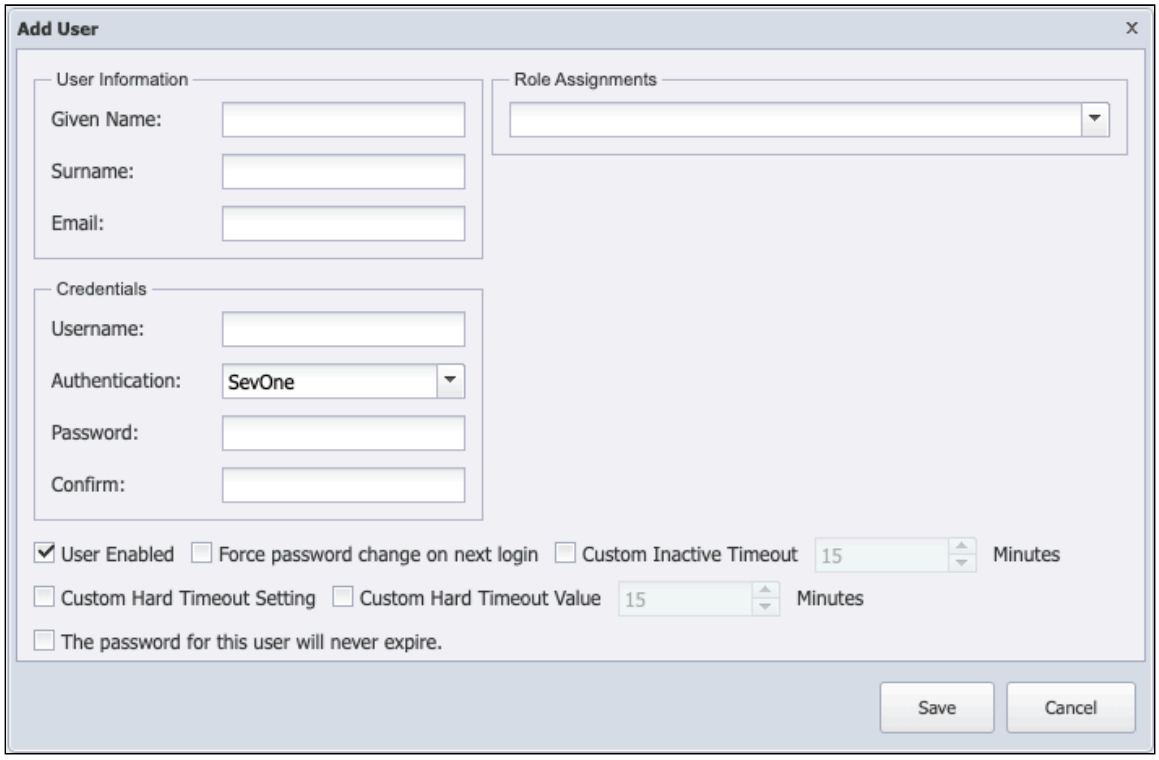
   a. Enable **Role View** to grant permission to view the name of the user role. Users cannot see users assigned to the user role until you enable applicable User View permissions.
   b. Enable **Role Edit** to grant permission to edit the user role name. Enable applicable User View permissions to grant permission to manage the user assignments for the role.
   c. Enable **User View** to grant permission to view the users in the role.
   d. Enable **User Edit** to grant permission to edit the user information, credentials, and role assignments for the users in the role.

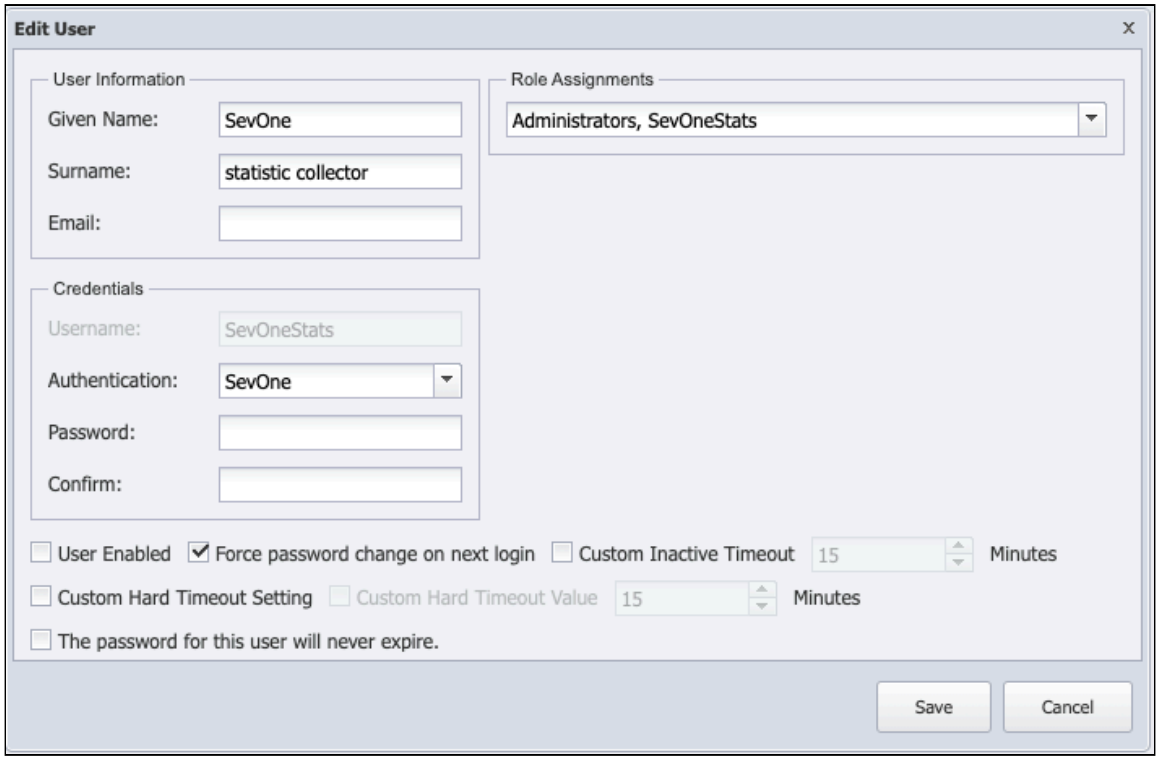
7.3 Users

The **Users** section on the lower-half of the page enables you to manage users and user role assignments. Users can update their given name, surname, email address, and password from the **Preferences** page.

1. Select the check box for each user to manage, click  , and select **Enable**, **Disable**, or **Delete** to enable, disable, or delete the users you select.
2. Click **Add User** or  to display the Add/Edit User pop-up that enables you to manage the user information, credentials, and role assignments.

Example: Add / Edit User





To edit an existing user, select the user and click to display the **Edit User** pop-up. While editing a user, username cannot be changed.

3. User Information
    a. **Given Name** - enter the given name to appear wherever a user name appears.

   b. **Surname** - enter the surname to appear wherever a user name appears.
   c. **Email** - enter the email address where you want SevOne NMS to send emails to the user.
4. **Credentials**
   a. **Username** - enter the name for the user to enter into the Username field on the Login page. You cannot edit this field after you click Save.
   b. **Authentication** - click drop-down and select the method for the user to use when they log on. Select the **SevOne** authentication unless your company uses LDAP, RADIUS, or TACACS protocol to authenticate users.
   c. **Password** - enter the user password. This field and the Confirm field are not applicable for users who use TACACS, LDAP, or RADIUS because password management for these protocols is done through the corresponding authentication servers.
   d. **Confirm** - re-enter the user password.
5. **Role Assignments** - click the drop-down and select the user roles to which to assign the user. You can assign users to multiple roles and role permissions are cumulative.
6. Select the **User Enabled** check box to enable the user to log on and use SevOne NMS. Clear this check box to block access to the user without having to delete their account.
7. Select the **Force password change on next login** check box to force the user to change the password when they log on for the first time.
8. Select the **Custom Inactive Timeout** check box to enable the user to stay logged on during periods of inactivity for the amount of minutes you enter in the Custom Inactive Timeout field. This setting overrides the **Inactivity Timeout** setting you enter on the Cluster Manager > **Cluster Settings** tab > **Security** subtab. Leave clear to have the user log off after the amount of time you enter on the Cluster Manager. The user must log out and then log back on for this setting to take effect.
9. Select the **Custom Hard Timeout Setting** check box to enable and use customized Hard Timeout setting (for the user you are adding or editing) you define on the Cluster Manager > **Cluster Settings** tab > **Security** subtab.
10. To customize the hard timeout value for a user, select the **Custom Hard Timeout Value** check box to enable editing the hard timeout value for the user that you are either adding or editing. Enable the checkbox to allow you to enter the number of minutes in value field the user can remain alive before SevOne NMS automatically logs them out of the application. The default value is 15 minutes. Value field can range between 5 minute to 86400 minutes (60 days). When Custom Hard Timeout Value is enabled, the timeout set in its value field is used for the user in add or edit mode instead of the Hard Timeout value set on the Cluster Manager > **Cluster Settings** tab > **Security** subtab.
11. Select the **The password for this user will never expire** check box to override the **Maximum Password Age** setting you define on the Cluster Manager > **Cluster Settings** tab > **Security** subtab. This check box does not appear when you do not enable the Maximum Password Age setting on the Cluster Manager.
12. Click **Save**.

> Your new user will also appear on **Administration** > **Access Configuration** > User Manager page.

7.4 Troubleshooting

I disabled a page permission but users assigned to the user role still have the user permissions associated with it.

A bit earlier, we mentioned that you will need to manually disable corresponding user permissions for a page permission when you disable that page permission. Otherwise, the corresponding user permissions remain enabled even after you have disabled the page permission.

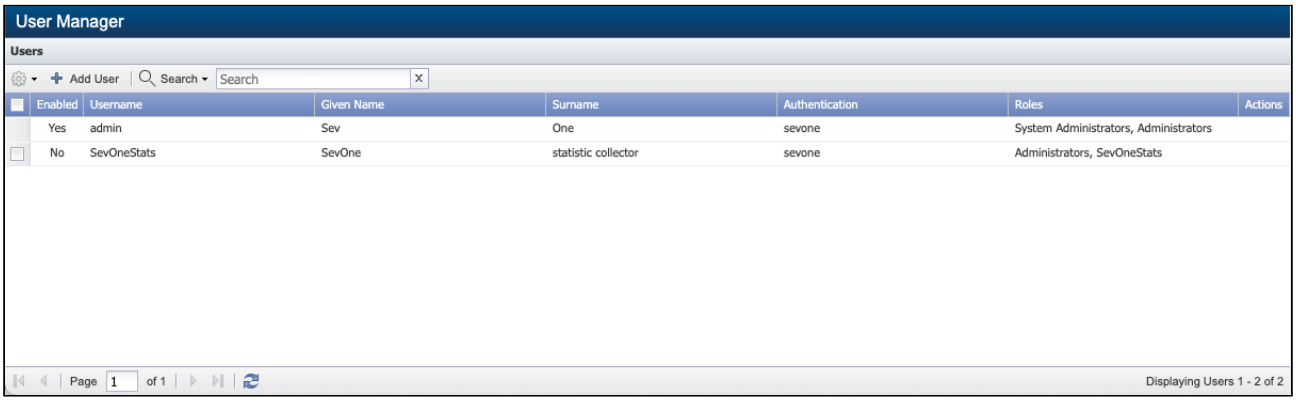
7.5 Terms

Lightweight Directory Access Protocol (LDAP)	An application protocol to query and modify directory services that run over TCP/IP to enable maintenance of centralized user directories that distributed applications authenticate to.
Remote Authentication Dial-In User Service (RADIUS)	A network protocol that provides centralized access, authorization, and accounting management for people or computers to connect and use a network service.
Terminal Access Controller Access Control System (TACACS)	A remote authentication protocol that communicates with an authentication server commonly used in UNIX networks.

8 User Manager

The User Manager enables you to manage user information, credentials, and user role assignments. You define user permissions and device permissions from the User Role Manager. When you are a user who is assigned to a role that has the Can Create Users User Permission, you can manage the information for the users that are assigned to the roles to which you have User Edit permission.

To access the User Manager from the navigation bar, click the **Administration** menu, select **Access Configuration**, and then select **User Manager**.



The users who are assigned to the user roles to which you have User Edit permission appear in the list.

- **Enabled** - Displays *Yes* if the user account is enabled and can use SevOne NMS or displays *No* if the user account is disabled.
- **Username** - Displays the user log on name.
- **Given Name** - Displays the user given name.
- **Surname** - Displays the user surname.
- **Authentication** - Displays how the user authenticates onto the application, either directly into *SevOne* or via *LDAP*, *RADIUS, or TACACS*.
- **Roles** - Displays the roles to which the user is assigned.

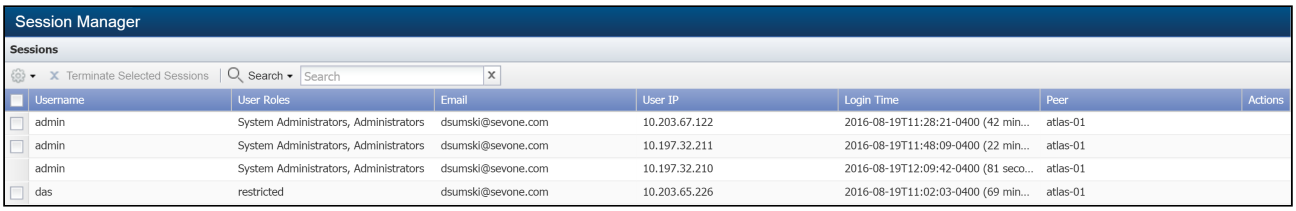
8.1 Users

Users can update their given name, surname, email address, and password from the Preferences page.

Please refer to User Role Manager > section **Users** for details on adding/editing a user.

9 Session Manager

The Session Manager allows you to view information about active sessions–for example, the user name and email address of anyone currently logged in to SevOne NMS–and to terminate sessions.



1. From the navigation bar, click **Administration** and select **Access Configuration**, then **Session Manager**.
2. The Session Manager provides the following information about active sessions:
    - **Username** - The user name of the person who is logged in to SevOne NMS.
    - **User Roles** - The user roles that apply to that user account.
    - **Email** - The email address associated with the user account.
    - **User IP** - The IP address of the device the user logged in from.
    - **Login Time** - The date and time that the user logged in to SevOne NMS. The current duration of the session appears in parentheses.
    - **Peer** - The peer that the user is logged in to.
3. To terminate a session, perform the following actions:
    a. Select the check box for the session and click on **Terminate Selected Sessions**. You may select more than one session.

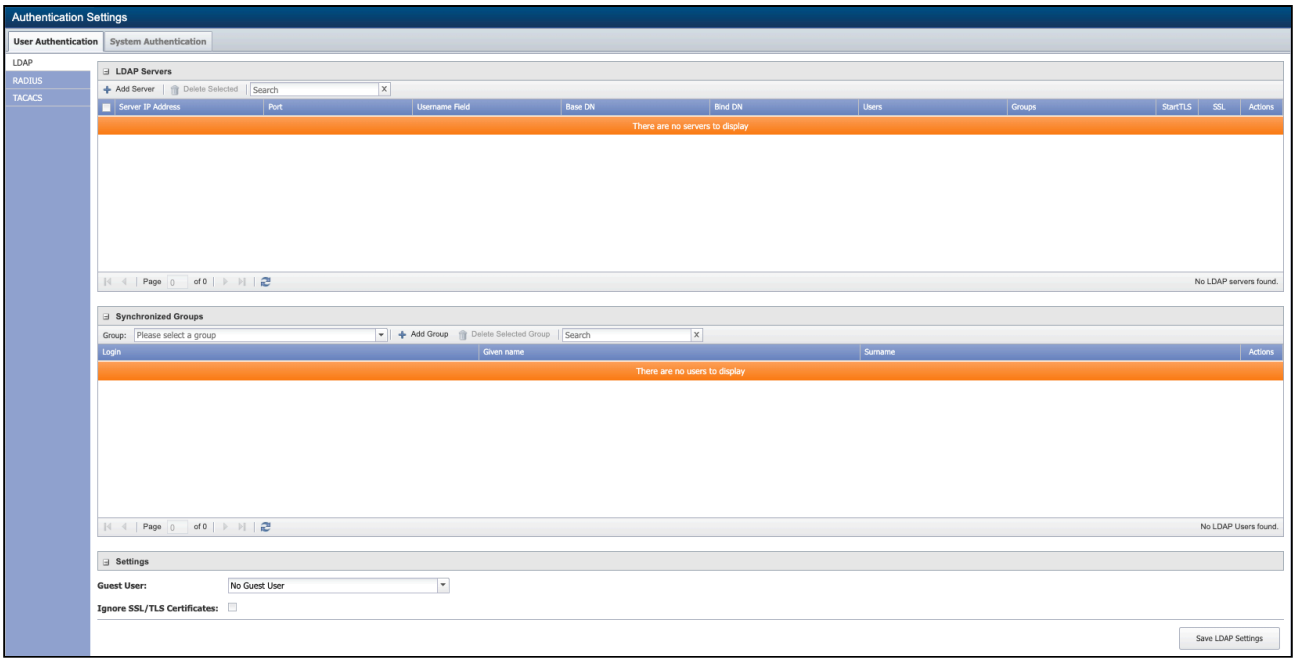
    > You will notice that one of the listed sessions does not include a check box. This is your active session.

    b. The **Confirm** pop-up appears. Click **Yes** to confirm that you would like to terminate the session. Otherwise, click **No** to cancel the operation.

10 Authentication Settings

The Authentication Settings page enables you to configure SevOne NMS users to access the application via LDAP, RADIUS, and TACACS protocol authentication. The System Authentication tab enables you to upload security certificates.

To access the Authentication Settings page from the navigation bar, click the **Administration** menu, select **Access Configuration**, and then select **Authentication Settings**.



10.1 User Authentication

The User Authentication tab enables you to configure SevOne NMS to use LDAP, RADIUS, and TACACS protocol authentication.

- **Lightweight Directory Access Protocol (LDAP)** - An application protocol to query and modify directory services that run over TCP/IP to enable maintenance of centralized user directories to which distributed applications authenticate.
- **Remote Authentication Dial In User Service (RADIUS)** - A network protocol that provides centralized access, authorization and accounting management for people or computers to connect and use a network service.
- **Terminal Access Controller Access-Control System (TACACS)** - A remote authentication protocol that communicates with an authentication server commonly used in UNIX networks.

10.1.1 LDAP

The LDAP subtab enables you to configure communication with the LDAP protocol authentication server.

LDAP refers to the Lightweight Directory Access Protocol. It is an industry standard application protocol for accessing and maintaining distributed directory information services over the IP network. Using LDAP, organizations can maintain centralized directories of users, groups, systems, networks, services, etc. Various distributed applications use LDAP to authenticate against those directories.

LDAP directories use a tree structure for storing information. This structure is known as a Directory Information Tree (DIT). The directory tree contains three main components:

- Trunk
- Branches
- Leaves

The trunk is the directory root. It will most likely be named after a domain. For example, if your domain is example.com, the root of your directory would be named dc=example, dc=com. The branches of the trunk are organizational units. If your organization has multiple sites, you might have an organizational unit, or '**ou**', for each site. For example, you could have one ou for California, another one for Texas, and another for Pennsylvania and as many ou's as you wish.

Just as an individual branch can have its own branches, an ou can have, or contain, its own ou's. The ou's mentioned above might each contain three subordinate ou's: Users ou, a Groups ou, and a Machines ou. These ou's can also contain ou's, but they do not have to. The Users ou, for instance, might just contain the users for that location. The actual user entries would be considered leaves because they cannot contain any subordinate entries.

A few benefits of the tree structure are:

- Increased ease of administration and maintenance
- Flexible application of security policies and access controls
- Scalability
- Simplified resource sharing

Common systems that provide implementations of LDAP include Microsoft's ActiveDirectory, the open source OpenLDAP project, and the Oracle Internet Directory product line.

SevOne NMS supports LDAP authentication for individual users and LDAP group synchronization for Active Directory and OpenSSL. Group synchronization occurs once per hour. A user group in LDAP creates a user role in SevOne NMS, however, manually adding a user to that role may result in automatically removing the added user from that role and/or deleting it from SevOne NMS.

Any LDAP authenticated user who has the Must Change Password at Next Logon (or similar) setting on the LDAP server and has NOT changed said password will NOT be able to log on to SevOne NMS. Either disable this setting for the user on the LDAP server or ensure that LDAP users change their passwords elsewhere before attempting to log on SevOne NMS.

When LDAP Group Synchronization is enabled, SevOne NMS attempts to synch LDAP users from any configured groups into the SevOne NMS user repository on an hourly basis. Relevant properties are populated per the following:

- givenname -> Given Name
- sn -> Surname
- mail -> Email

Perform the following steps to manage LDAP authentication.

10.1.1.1 LDAP Servers

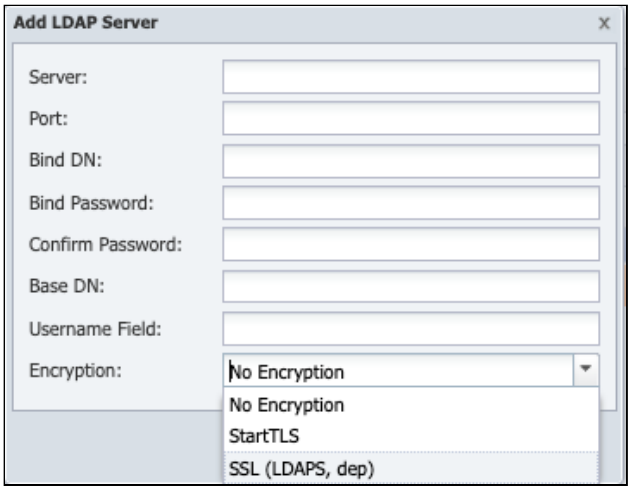
SevOne NMS maintains consistency between the remote LDAP server and the synced local users who have only an LDAP role. This means that when such a user is removed from the remote LDAP server, SevOne NMS also removes the corresponding local user.

1. In the LDAP Servers section, click **Add Server** above the server list or click to display the Add/Edit LDAP Server pop-up.

a. In the **Server** field, enter the host name or IP address of the LDAP server.
b. In the **Port** field, enter the network port of the LDAP server. The default LDAP port is 389. The default LDAPS port is 636 which has been deprecated.
c. In the **Bind DN** field, enter the name of the user SevOne NMS is to use to authenticate to the directory. This is the username that is authorized to perform searches within the context of the Base DN in the previous step, which means that the bind DN's authorizations also allow SevOne NMS to search the directory tree.

> **Example:** *cn=guest*

d. In the **Bind Password** field, enter the password for the user name you enter in the previous step. This is not required in LDAP version 3 (LDAPv3).
e. In the **Confirm Password** field, reenter the bind password.
f. In the **Base DN** field, enter the base distinguished name (DN) on which to do LDAP queries. The top level of the LDAP directory tree is the base, referred to as the *base DN* from which a search starts. For an Active Directory system this is typically *dc=example, dc=com*.
g. In the **Username Field**, enter the LDAP field SevOne NMS is to check to find user names. In Active Directory, this is typically *sAMAccountName*. Many other directories use *cn* or *uid*.
h. Click the **Encryption** drop-down.



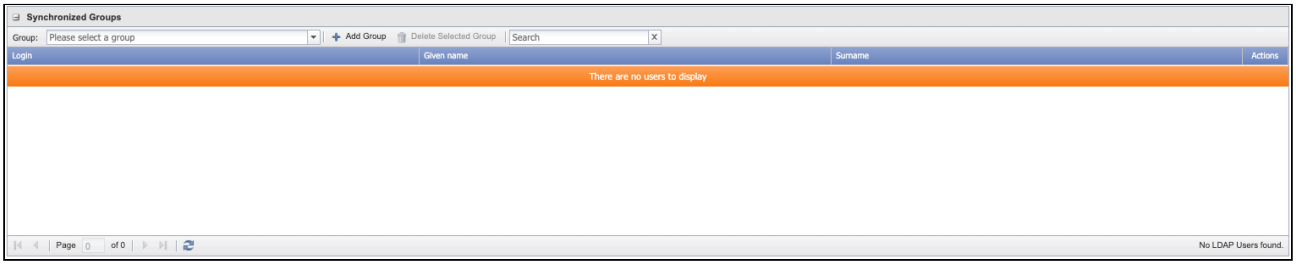
- Select **No Encryption** to not use encryption.

- Select **StartTLS** to use StartTLS. StartTLS secures the LDAP credentials and data. StartTLS is sometimes referred to as the TLS upgrade operation because it upgrades a normal LDAP connection to a connection that is protected by TLS/SSL.
- Select **SSL (LDAPS, dep)** to use Secure Socket Layers (SSL). SSL secures LDAP data. A method to secure LDAP communication is to use an SSL tunnel. This is denoted in LDAP URLs by using the URL scheme "ldaps". The use of LDAP over SSL was common in LDAPv2. This usage has been deprecated along with LDAPv2.

i. Click **Save**.

2. Repeat to add additional servers.
3. In the server list, the **StartTLS** column and the **SSL** column enable you to change the related settings.
4. Click in the Actions column to test the connection to the LDAP server.

10.1.1.2 Synchronized Groups



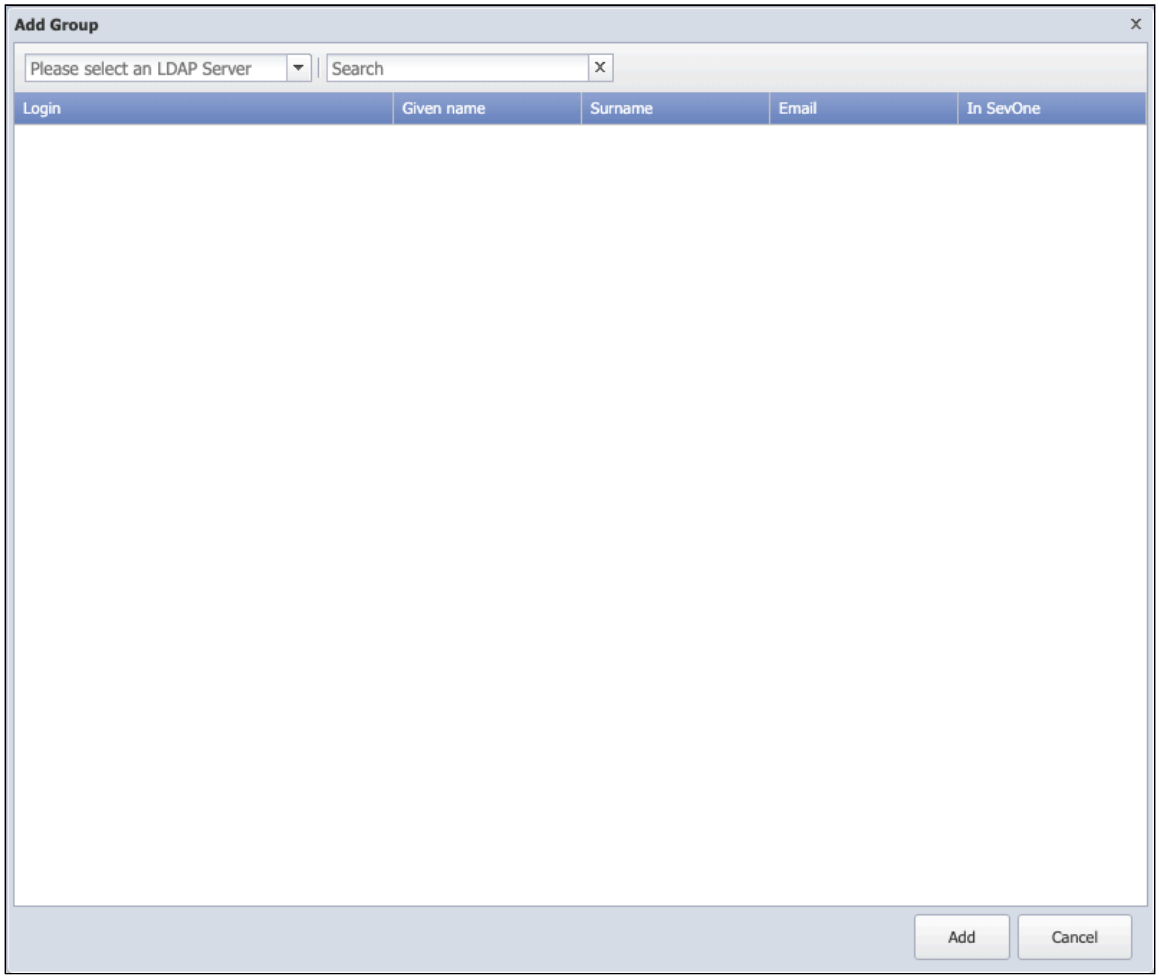
In the Synchronized Groups section, click the **Group** drop-down and select the server group to which to associate the server you select. LDAP groups are the equivalent of SevOne NMS user roles.

LDAP groups are associated with SevOne User Roles nested in the LDAP folder. The LDAP sync process will automatically perform the following actions:

- Create or delete User Roles within the LDAP folder hierarchy for any LDAP groups present during the sync.
- Create new user accounts for any users present in the LDAP groups.
- Add or remove User Roles to individual user accounts based on their LDAP group assignment.

LDAP roles created by the sync will have no permissions by default and must be maintained manually. If LDAP group assignment is changed for a user, the next LDAP sync will modify the user's roles in the NMS accordingly.
User roles not nested within the LDAP roles folder can be assigned to LDAP users but require manual management by an administrator.

1. If the group you are looking for does not appear, click **Add Group** to display the Add Group pop-up.

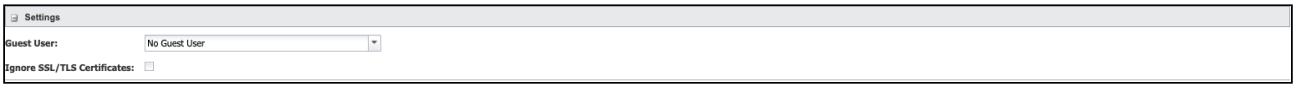


a. Click the **LDAP Server** drop-down and select a server.
b. In the **Search** field, enter at least one letter to filter the search results and press **Enter**.
c. In the list of groups, click the **+** next to the group name to display the group members.
d. Select the check box for each group to add.
e. Click **Add** to add the groups you select.

2. Click on **Delete Selected** to remove the group that is currently displayed in the **Group:** input box. Use the down arrow to select any group you wish to delete. All users that are only assigned to this group will be deleted. Users that have other group memberships will be retained.

10.1.1.3 Settings

In the Settings section, click the **Guest User** drop-down and select the guest user to provide permissions for anyone who logs on with a valid LDAP ID but no SevOne NMS account.



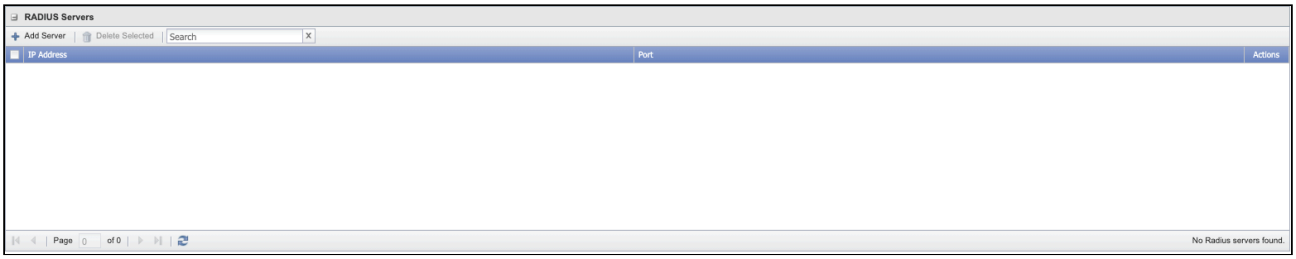
1. Select the **Ignore SSL/TLS Certificates** check box to skip verification of the server (not recommended). If you change this setting you must contact SevOne Support for it to properly take effect.

Click **Save LDAP Settings**.

10.1.2 RADIUS

The RADIUS tab enables you to configure SevOne NMS to communicate with the RADIUS protocol authentication server.

10.1.2.1 RADIUS Servers



1. Click **Add Server** above the server list or click  to display the Add/Edit RADIUS Server pop-up.

   a. In the **IP Address** field, enter the IP address for the RADIUS server.
   b. In the **Port** field, enter the RADIUS sever port number.
   c. In the **Shared Secret** field, enter the RADIUS server shared secret.
   d. Click **Save**.
2. Repeat to add additional servers.

10.1.2.2 Settings

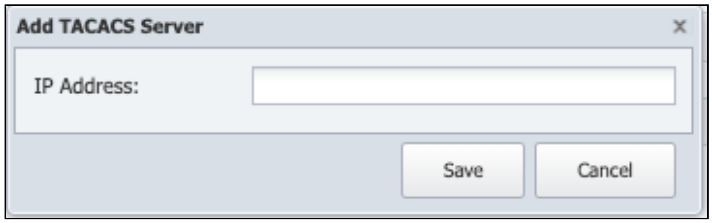
1. Click the **Encryption** drop-down and select the type of encryption to use.
2. Click the **Guest User** drop-down and select the guest user to provide permissions for anyone who logs on with a valid RADIUS ID but no SevOne NMS account.
3. In the **RADIUS NAS Identifier** field, enter the RADIUS NAS identifier, if required (default - localhost if left blank).
4. In the **RADIUS Calling Station ID** field, enter the RADIUS calling station identifier, if required (default - 127.0.0.1 if left blank).
5. Click **Save RADIUS Settings**.

10.1.3 TACACS

The TACACS subtab enables you to configure SevOne NMS to communicate with the TACACS protocol authentication server. The servers in the list are tested in the sequence in which they appear in the list. If the first server is running and the user does not have the proper credentials, then the user cannot log on. If that server is not running then the second server in the list attempts to log the user on.

10.1.3.1 TACACS Servers

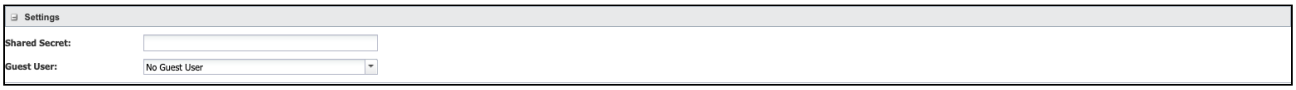
1. Click **Add Server** above the server list or click  to display the Add/Edit TACACS Server pop-up.



a. In the **IP Address** field, enter the IP address of the TACACS authentication server.
b. Click **Save**.

2. Repeat to add additional servers.

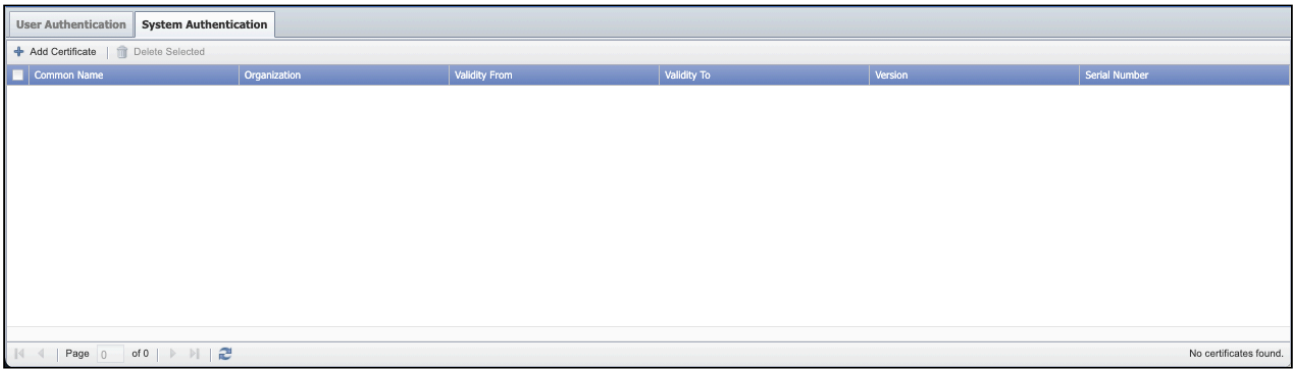
10.1.3.2 Settings



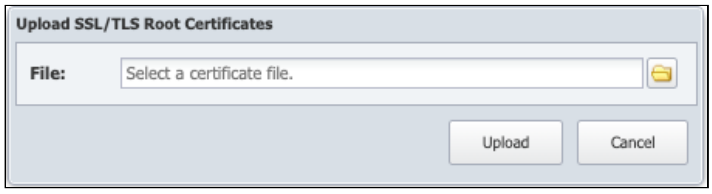
1. In the **Shared Secret** field, enter the shared secret for the server.
2. Click the **Guest User** drop-down and select the user to provide permissions for anyone who logs on with a valid TACACS ID but no SevOne NMS account.
3. Click **Save TACACS Settings**.

10.2 System Authentication

The System Authentication tab enables you to upload security certificates. SevOne NMS uses authentication certificates for LDAP. You also need to upload a certificate if you want to use the HTTP plugin and/or the Web Status plugin with a log on via https. You must upload the CA Root Certificates to enable SevOne NMS to communicate with an LDAP server that has certificates that are signed by an unknown CA. The certificates must be base64-encoded PEM files. It can take up to fifteen minutes for a certificate to synchronize across your SevOne cluster.



1. Click **Add Certificate** to display the Upload SSL/TLS Root Certificate pop-up.



2. Click  to locate and select the certificate.
3. Click **Upload** to upload the certificate.

Certificate Information

The System Authentication tab provides the following information for certificates that have been uploaded.

- **Common Name** - The hostname that the certificate is associated with.
- **Organization** - The organization that the certificate is associated with.
- **ValidityFrom** - The date and time from which the certificate is valid.
- **ValidityTo** - The date and time at which the certificate stops being valid.
- **Version** - The certificate version number.
- **SerialNumber** - The certificate's serial number.

10.3 Troubleshooting

10.3.1 Check the basics.

It is a good idea to start with the basics. If you have added a server and the connection test has failed, double-check that all the basic settings are correct. Select the server in question and click  to display the Edit LDAP Server pop-up. Check line-by-line to confirm that the information in the fields is accurate. If you used a bind password, try reentering it to see if that fixes the problem.

10.3.2 You uploaded a certificate, and the server connection test fails when using StartTLS or SSL.

There are a few possibilities here:

1. The newly uploaded certificate may not have taken effect yet. Just give it a few minutes and try again. If that does not work, proceed to the next step.
2. Something went wrong during the upload. Try uploading your certificate again and wait about five minutes for it to take effect. If that does not work, proceed to step 3.
3. There is a problem with your certificate. If there is a problem with the certificate itself, you may need to get another copy of the certificate file. Upload the new certificate file and wait about five minutes for it to take effect. If you are still having problems after that, the original certificate file may be corrupted. If it is, you will need to get a good certificate file and upload that. Once again, give it about five minutes to take effect.

10.4 Terms

Authentication	The process of verifying that someone is who they claim to be.
Authorization	The process of allowing someone access or information.
Certificate	A file used to verify that its owner (for example, a server) is who it says it is.
Certificate Authority (CA)	A trusted third party that issues digital certificates, which certify that the certificate owners are who they say they are.
Encryption	The process of converting data into a format that can only be read by authorized users.

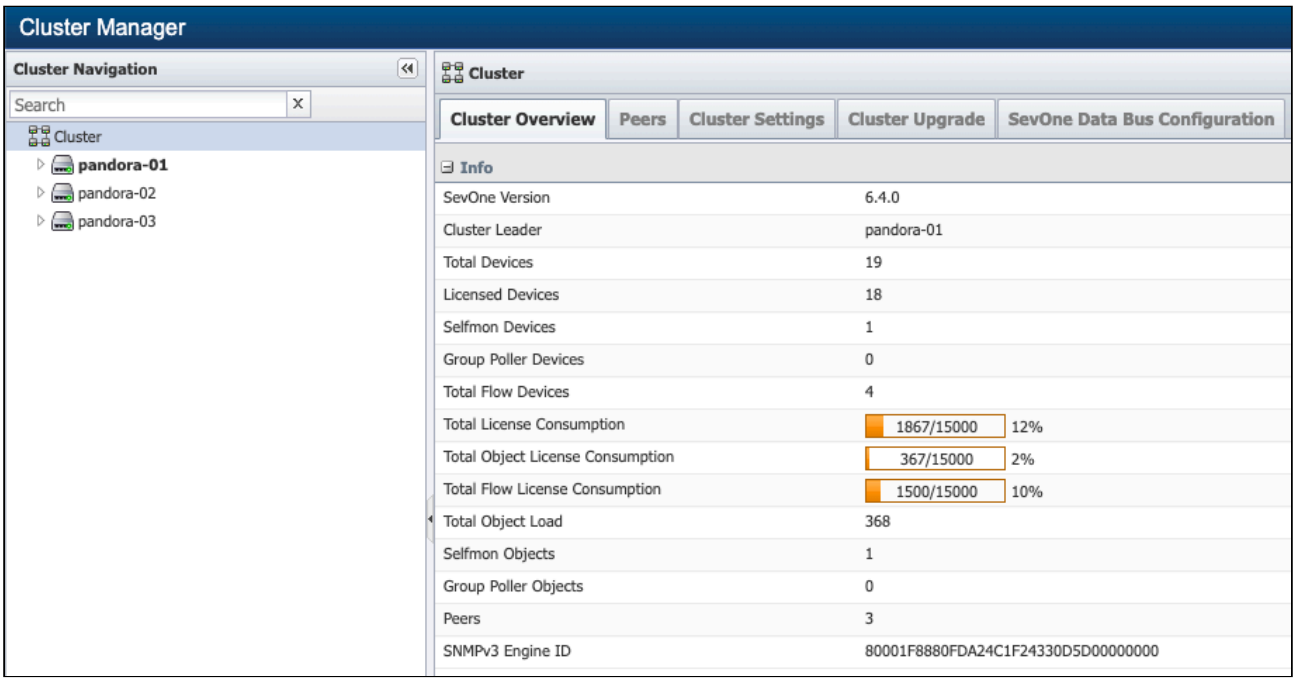
11 Cluster Manager

The Cluster Manager displays statistics and enables you to define settings at the cluster level, the peer level, and the appliance level. With a few exceptions, the default Cluster Manager settings enable you to run SevOne NMS right out of the box. The Cluster Manager also enables you to integrate additional SevOne NMS appliances into your cluster.

To access the Cluster Manager from the navigation bar, click the **Administration** menu and select **Cluster Manager**.
The following Cluster Settings are specific to your network.

- Devices - Device name masks
- *Devices - Time zone
- *Email - Your network's email server specifications
- SFTP - Your network's SFTP specifications
- *SNMP - Community strings

*You probably defined these settings from the Startup Wizard upon initial implementation.



The left side enables you to navigate your SevOne NMS cluster hierarchy to view statistics and define settings at the cluster level, the per level, and the appliance level. When the Cluster Manager appears, the default display is the cluster level with the **Cluster Overview** tab selected.

- **Cluster Level** - The cluster level enables you to view cluster-wide statistics, to view statistics for all peers in the cluster, and to define cluster-wide settings.
- **Peer Level** - The peer level enables you to view peer specific information and to define peer specific settings. In the cluster hierarchy, the cluster leader peer name displays first in **bold** font and the other peers display in alphabetical order.
- **Appliance Level** - Each Hot Standby Appliance peer pair displays the two appliances that act as one peer in the cluster. The appliance level enables you to view database replication details, to configure settings to meet Common Criteria security standards, to manage application processes, to view system logs, to add a new peer to your cluster, etc.

11.1 Cluster Level Options

Cluster - When you select the cluster level in the hierarchy on the left, the following tabs appear on the right to enable you to view cluster level information and to define cluster level settings.

- **Cluster Overview** - Enables you to view cluster-wide information.
- **Peers** - Enables you to view the list of peers in the cluster with peer statistics.
- **Cluster Settings** - Enables you to define cluster-wide settings across all peers in the cluster.

- **Cluster Upgrade** - Enables you to upgrade the artifact via the SFTP server, run the installer to use the newly downloaded Upgrade Artifact, and view the URL. Also, it shows the cluster upgrade history. This tab appears on the **active** Cluster Leader only.
- **SevOne Data Bus Configuration** - Enables you to configure SevOne Data Bus using the Graphical User Interface.

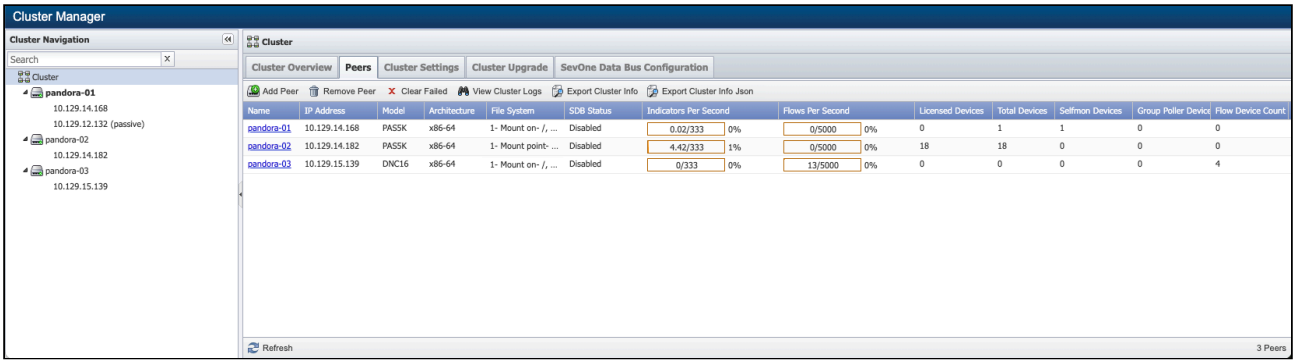
11.1.1 Cluster Overview

Click **Cluster** in the cluster hierarchy on the left and select the **Cluster Overview** tab on the right to display cluster-wide information that includes the total objects and flow load statistics that enable you to see how much of your license object capacity your cluster uses.

- **SevOne Version** - Displays the SevOne NMS software version.
- **Cluster Leader** - Displays the name of the cluster leader peer. The configuration settings such as cluster settings, security settings, device lists, etc. you define from any peer are stored in the config database on the cluster leader peer. All active peers in the cluster pull config database changes from the cluster leader peer.
- **Total Devices** - Displays the total number of Licensed, Selfmon, and Group Poller devices in the cluster.
- **Licensed Devices** - Displays the number of devices in your network that SevOne NMS has discovered from which objects are capable of being polled. The Device Manager enables you to manage devices. The Licensed Devices count is equal to (Total Devices - (Selfmon Devices + Group Poller Devices)) in the cluster. For details, please refer to section **Device Manager** in *SevOne NMS User Guide*.
- **Selfmon Devices** - Displays the number of Selfmon devices in the cluster.
- **Group Poller Devices** - Displays the number of Group Poller devices in the cluster.
- **Total Flow Devices** - Displays the total number of flow devices in the cluster.
- **Total License Consumption** - Displays the sum usage of objects and flow. This displays the number of flows and objects the cluster is licensed to use and the percentage of the license capacity your cluster uses.
- **Total Object License Consumption** - Displays total usage of objects. This displays the number of objects the cluster is licensed to use and the percentage of the license capacity your cluster uses.
- **Total Flow License Consumption** - Displays the sum usage of flows. This displays the number of flows the cluster is licensed to use and the percentage of the license capacity your cluster uses.
- **Total Object Load** - Displays the total number of objects polled from all peers in the cluster along with Selfmon and Group Poller Objects. The Object Types page, Object Rules page, and Object Manager (please refer to *SevOne NMS User Guide* for details) enable you to manage the number of polled objects.
- **Selfmon Objects** - Displays the number of Selfmon objects in the cluster.
- **Group Poller Objects** - Displays the number of Group Poller objects in the cluster.
- **Peers** - Displays the number of peers in the cluster.
- **SNMPv3 Engine ID** - Display the Engine ID of the SevOne NMS cluster.

11.1.2 Peers

Click **Cluster** in the cluster hierarchy on the left and select the **Peers** tab on the right.



11.1.2.1 Add Peer

Allows you to queue the new peer for integration, then click Add Peer to add the new peer to the cluster. This works in conjunction with the **Integration** tab found at the appliance level on the Cluster Manager. When you want to add a new appliance to your cluster as a new peer, you start the integration process on the new appliance and finish the integration process on an appliance that is already in the cluster.

Please refer to the Integration section below for instructions on how to add a new peer to your cluster.

11.1.2.2 Remove Peer

Caution should be used when considering the use of this feature.

> **IMPORTANT**
>
> When a peer is removed, **/etc/hosts** file is recreated. This results in user losing all the host entries in the hosts file. Please proceed with caution!

> You must contact SevOne Support to re-add the peer to the cluster.
> Remember: The Add Peer button removes all existing data on the appliance.

This feature should only be considered when the peer is acting in a way that is adverse to the overall cluster functionality or performance. In a Hot Standby Appliance peer pair, if you click this button you remove both appliances in the peer.

The following statements assume that the peer is still functional enough to continue to appropriately run the SevOne NMS software.

- All devices polled by the peer are not removed from the peer and are not distributed to other peers. These devices are inaccessible from the peers that remain in the cluster.
- Data is not removed from the peer you remove from the cluster.
- After you agree to the confirmation prompts, there is no way to cancel the peer remove process.
- The removed peer no longer appears in the hierarchy on the Cluster Manager.
- The peer removal is bi-directional which means the removed peer is excised from the net.peers table and the removed peer attempts to change its cluster leader to itself. If the removed peer is partially or totally unresponsive, this function restricts MySQL access to remove the affected peer from the cluster leader.

11.1.2.3 Clear Failed

Allows you to remove peer add failure messages from the list.

11.1.2.4 View Cluster Logs

Allows you to view the logs of the integration messages.

11.1.2.5 Export Cluster Info

Allows you to export *cluster* info to a **.csv** file.

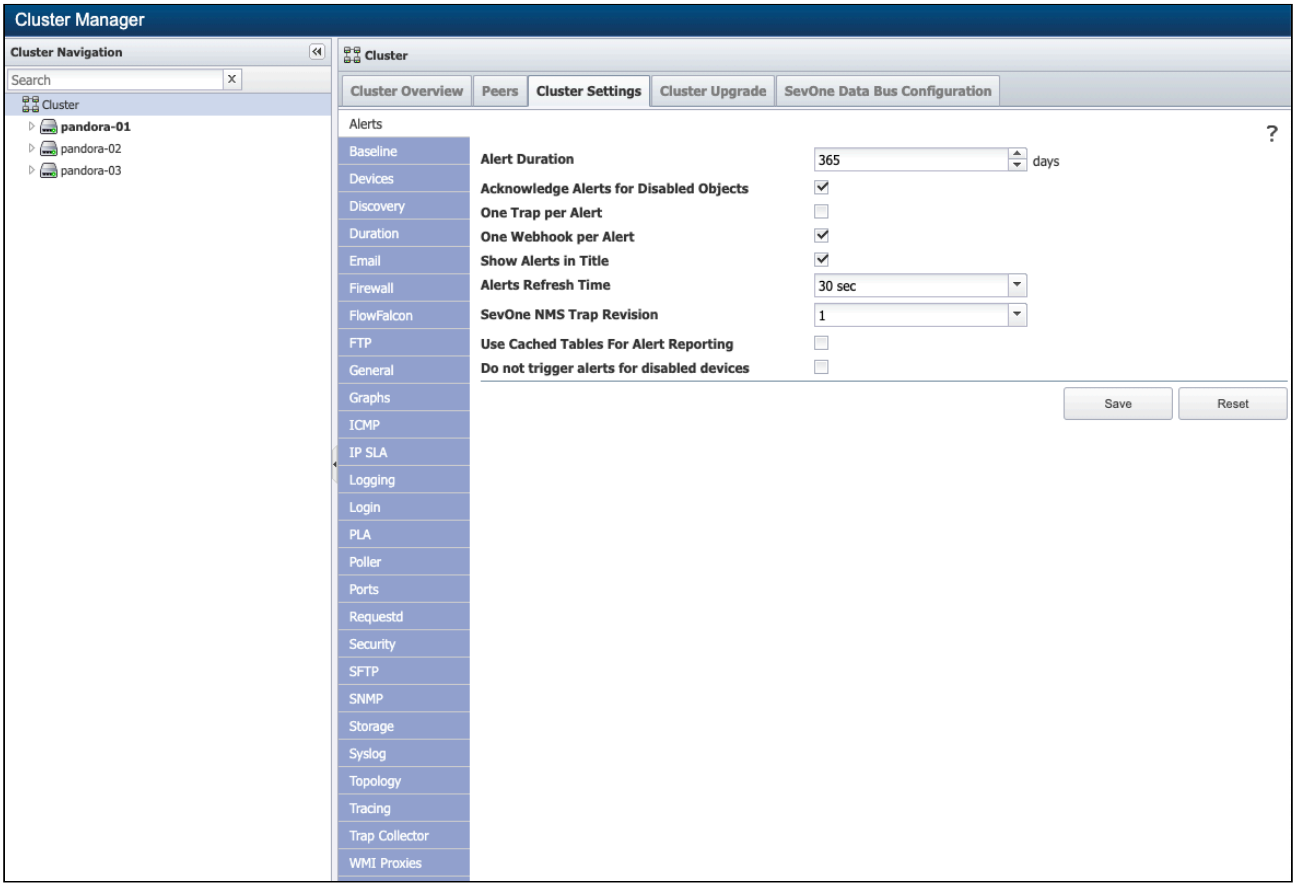
> **Export peer info for the selected peer only**
>
> To export peer information for the selected peer only, from **Actions** column, click  to **Export Peer Info**. This will export the information for the *selected peer only* to a **.csv** file.

11.1.2.6 Export Cluster Info Json

Allows you to export *cluster* information in gzipped JSON format. For example, *<clusterName>-<epochTimestamp>-ClusterInfo.gz*.

11.1.3 Cluster Settings

Click  **Cluster** in the cluster hierarchy on the left and select the **Cluster Settings** tab. Subtabs appear along the left side of the Cluster Settings tab to enable you to define cluster level settings.



11.1.3.1 Alerts

The Alerts subtab enables you to define alert settings that affect Alerts page and related pages workflows. For details on **Alerts**, please refer to *SevOne NMS User Guide*.

1. In the **Alert Duration** field, enter the number of days' worth of alert information to store (between 0 and 365). The default is 365.

   > SevOne recommends the alert archives are less than 2 million alerts. To trim, modify the Alert Duration field or please contact **SevOne Support** for help.

2. Select the **Acknowledge Alerts for Disabled Objects** check box to clear alerts for objects that you disable.
3. Select the **One Trap per Alert** check box to send only one trap per alert. Leave clear to send a trap every time a threshold triggers, even if an alert already exists.
4. Select the **One Webhook per Alert** check box to send only one webhook per alert. Leave clear to send a webhook every time a threshold triggers (on every occurrence).
5. Select the **Show Alerts in Title** check box to display the number of alerting devices and the highest alert severity level in the web browser tab.
6. Click the **Alerts Refresh Time** drop-down and select the frequency at which to refresh the Alerts page display. For large clusters that trigger many alerts, you should consider setting this higher than the 30 second default setting. The Alert engine runs every three minutes to trigger alerts and you cannot configure the alerts engine via the UI.
7. Click the **SevOne NMS Trap Revision** drop-down and select **1** for revision one traps or select **3** for revision three traps or select **4** for revision four traps. This defines what trap data SevOne NMS sends to your fault management system. If you change this you will need to update how your fault management system receives traps from SevOne NMS. Please refer to section Trap Revisions for details.
8. Select the **Use Cached Tables For Alert Reporting** check box to use a cached version of the table to reduce read contention.
9. Select the **Do not trigger alerts for disabled devices** check box to prevent alerts from being generated for devices that polling has been disabled for (on the Device Manager page).
10. Click **Save** to save the Alerts settings.

11.1.3.2 Baseline

The Baseline subtab enables you to define how to create baselines. Baselines are used in report workflows and policy/threshold workflows. For details, please refer to sections **Report Manager** and **Policy Browser** in *SevOne NMS User Guide*.

Exponential Smoothing: SevOne NMS uses a rolling average formula known as Exponential Smoothing for baseline calculation. This uses the scalar value of the previous average and the newly collected value to compute the new average. There is no reliance on the actual data points collected during the previous <n> weeks.

> **Example**
>
> - Baseline Weight (weight of the existing baseline data) = 10
> - New Data Weight (weight of new baseline data) = 1
> - baseline = (existing baseline * 10 + new value * 1)/ (10 + 1)
>
> baseline = new value if there is no existing baseline

The value of (old weight / (old weight + new weight)) is the smoothing factor. The smoothing factor affects the resistance to change that new data has on a baseline. This value ranges between 0 and 1 and a higher smoothing factor causes a greater resistance to change.

> Back to the example:
> The default smoothing factor is 10/11 ~= 0.909.

As the smoothing factor approaches 1, the impact of each new value on the existing baseline approaches zero. This approach calculates the average and the standard deviation. No trending (slopes) is considered in the calculation as each baseline data point is computed individually.

> Changes to these settings can cause data loss. From a data perspective, altering Baseline settings is extremely dangerous. After you change the granularity and click Save, you destroy ALL current baseline data across all peers.

1. In the **Granularity** field, choose the granularity of a baseline in seconds from the drop-down. The default is 900 seconds (15 minutes) which takes all data during a 15 minute time span, averages the 15 minutes' worth of data, and stores that average data point for every 15 minutes of the week for a total of 672 data points in order to create baselines. The minimum value is 240 seconds (4 minutes) and maximum value is 3600 seconds (60 minutes or 1 hour).
2. In the **Baseline Weight** field, enter the weight to apply to existing baselines (between 1 and 52). The default is 10. A larger number here reduces the impact of new data on the baseline.
3. In the **New Data Weight** field, enter the weight to apply to new data (between 1 and 52). The default is 1. A larger weight here causes new data to change the baseline faster.
4. Click **Save** to save the Baseline settings.

11.1.3.3 Devices

The Devices subtab enables you to define device definition settings for the Device Manager and related workflows. For details, please refer to section **Device Manager** in *SevOne NMS User Guide*.

1. Select the **Prevent Duplicate IP Addresses** check box to prevent the addition of a device with an IP address that is already in SevOne NMS.
2. Select the **Discover Trap Destinations** check box to discover trap destinations on devices.
3. Click the **Propagate child rules up to the parent** drop-down and select one of the following options. The default is **Prompt**.
   - **Don't allow** - Do not allow child rules to propagate up to the parent.
   - **Prompt** - Will prompt you whether to allow child rules to propagate up to the parent.
   - **Automatically** - Automatically propagate child rules up to the parent.
4. In the **Device Names** section:
   a. Select the **Resolve Device Names** check box to update the device IP address to the resolved address. If you do not enter the correct IP address when you add a device (or an IP address changes) and DNS can resolve the device's name, the device IP address in SevOne NMS updates upon discovery.
   b. Select the **Lookup Hostnames** check box to rename devices whose names are IP addresses to their hostname. If you enter an IP address as a device name and DNS can resolve the IP address, the device name in SevOne NMS changes from the IP address to the device's hostname upon discovery.
   c. Select the **Lookup SysNames** check box to rename devices whose names are IP addresses to their sysName. If you enter an IP address as a device name and DNS cannot resolve the device name or you do not select the Lookup

Hostnames check box, the device name in SevOne NMS changes to the SNMP sysName upon discovery, if possible. The device name updates during discovery only if the current name is an IP Address. This check box does not cause the device name to change in SevOne NMS if you change the sysName on the device, in which case you must manually change the device name in SevOne NMS.

d. Select the **Force Hostnames** check box to use DNS to change all device names to their host names if DNS can resolve the device name.

5. In the **Device Name Masks** section, view the device name masks you define to mask (hide) device names.
   a. Click **Add** to add a row to the list.
   b. In the text field, enter a valid Perl regular expression.
   c. Click **Update**.
   d. Repeat to define the list of expressions.
   e. Click the arrows to move the expression up or down in the list to arrange the sequence of expressions. The mask process stops when a match is found.
6. In the **Device Deletion Queue Information** section:

> **About Device Deletion Queue**
>
> The Device Deletion Queue is a tool to safeguard device data from accidental deletion. By hiding & disabling devices that are added to the queue, devices are effectively removed from service without their historical data being deleted. When devices are deleted accidentally or unexpectedly, the absence of these devices from reports and alerts is often enough for users who rely on them to contact an administrator to communicate that there is a problem. An administrator may remove devices from the deletion queue to return them into service. While historical data, alerts and Device & Object group references to these devices in reports are maintained, there will be a data gap between the time the device was put into the queue and the time it was removed. Administrators may expedite deletion of devices from the queue on a per-device basis or by reducing the number of days devices are maintained in the queue before their permanent deletion.

   a. Click the **Days to delay** drop-down to select the timespan to delay the deletion of the devices in the queue by the number of days entered in this field. The default value is **0 days**. The minimum value is 0 days and maximum value is 31 days.

> If **Days to delay** field is set to **0 days**, then the device(s) in the deletion queue are marked for immediate deletion. This is to preserve the pre-existing behavior and allow the feature to be turned off.

   b. Select the **Disable Devices** check box to disable objects, polling, and alerting for the devices in the deletion queue.

> This setting applies only for the new devices added to the device deletion queue.

   c. Select the **Hide Devices** check box to hide devices queued for deletion from various reports such as, Device, Metadata, Topology, Performance Metrics, TopN, etc., in the user interface. These devices are not visible from Report Manager or Instant Graphs but their device summary is visible.
7. In the **Time Information** section:
   a. Click the **Default Date Format** drop-down and select the date format to use by default. Each user can override this setting from the Preferences page. For details, please refer to section **Preferences** in *SevOne NMS User Guide*.
   b. Click the **Default Time Zone** drop-down and select the time zone to appear by default in all device specific Time Zone fields.
   c. In the **Time Zone Filter** field, select the check box next to each country for which you want time zones to appear available for selection from the Time Zone drop-down lists. You must select at least one country time zone.
8. In the **Device Mover Settings** section:
   a. Select the **Source & Destination Health Check** check box to check the health of source and destination peers before the device is moved.
   b. Select the **Device Connectivity From Destination** check box to check the connectivity of the moving device from the destination peer before the actual move is performed.
9. Click **Save** to save the Device settings.

11.1.3.4 Discovery

The Discovery subtab enables you to define the way device discovery works to find the objects to poll.

1. Click the **Device Note Severity Level** drop-down and select the severity level at which to create device notes during discovery. For details, please refer to section **Discovery Manager** in *SevOne NMS User Guide*.

2. Click the **New Device Load Distribution** drop-down and select from options **Object** or **IPS** (Indicators Per Second). The default option is **Object**.
    a. The new devices created with auto peer are distributed based on the option selected from the drop-down list and automatically assigned to the peer with the least load (however, it will not be assigned to a DNC). From **Cluster Manager,** click  **<peer name>,** Peer Overview tab provides details for each peer available in the cluster.

    > **When importing from CSV,**
    >
    > From **Devices** > **Device Manager** > click **Import CSV** button > if field **applianceName** is empty, a peer is automatically chosen for the new device based on the option chosen from the drop-down list (Object or IPS).

    b. To determine if the peer is *full*,
        - if **New Device Load Distribution** is set to **Object**, it checks if the object and flow counts of all devices exceeds the peer capacity.
        - if **New Device Load Distribution** is set to **IPS**, it checks if the indicators per second in the past 2 hours on all the devices exceeds the peer capacity.

    If a new device is added and the peer load is equal to or greater than its maximum peer capacity, the **Objects** column under **Devices** > **Device Manager** displays *Peer Full*.
3. In the **Thread Pool** section:
    a. In the **Low Priority Size** field, enter the number of low priority devices to simultaneously discover (between 1 and 100). Automatic discovery is low priority. The default is 3, which is usually ideal for most implementations. There is a maximum thread pool of 100 devices that can be simultaneously discovered. More threads use more CPU and RAM, so you should reduce this number and/or the High Priority Size number if you notice system slow down.
    b. In the **High Priority Size** field, enter the number of high priority devices to simultaneously discover (between 1 and 100). The default is 3. User initiated discovery is high priority.
4. In the **Missing Objects** section:

    > When an indicator is discovered and you disable that Indicator Type from **Administration** > **Monitoring Configuration** > **Object Types**, the setting that determines when it will be removed from the report creation selection (for example, choosing indicators for an object in the Performance Metric Graph) is the **Days Until Delete** field.
    > Although the setting is for missing objects, the same applies for its indicators. If you add a new device and you have already disabled the Indicator Type from **Administration** > **Monitoring Configuration** > **Object Types**, the new device will not discover this indicator and it will not be available in the Instant Graph selection under your object.

    a. In the **Days Until Disable** field, enter the number of days to wait before an object that is not found during a *successful plugin-specific* discovery is marked disabled (between 0 and 9999). The default is 2. Enter 0 (zero) to disable missing objects as soon as SevOne NMS determines an object is missing.
    b. In the **Days Until Delete** field, enter the number of days to wait before an object that is not found during a *successful plugin-specific* discovery is deleted (between 0 and 9999). The default is 31. Enter 0 (zero) to delete missing objects (and all associated data) as soon as the object is determined to be missing. The value you enter in the **Days Until Delete** field must be greater than the value you enter in the **Days Until Disable** field.

    > Individual xStats indicators that have stopped transmitting data are subject to be disabled and deleted pending the **Days Until Disable** and **Days Until Delete** field settings. Previously, all indicators would remain regardless of their individual status as long as their object had any activity.

5. In the **Administrative Status** section:
    a. In the **Enable Up Objects** field, enter the number of days to wait before an object that is administratively up is enabled (between 0 and 9999). The default is 0. Enter 0 (zero) to not use this feature.
    b. In the **Disable Down Objects** field, enter the number of days to wait before an object that is administratively down is disabled (between 0 and 9999). The default is 0. Enter 0 (zero) to not use this feature.
6. In the **Operational Status** section:
    a. In the **Enable Up Objects** field, enter the number of days to wait before an object that is operationally up is enabled (between 0 and 9999). The default is 0. Enter 0 (zero) to not use this feature.
    b. In the **Disable Down Objects** field, enter the number of days to wait before an object that is operationally down is disabled (between 0 and 9999). The default is 3. Enter 0 (zero) to not use this feature.
    c. Select the **Preserve Max Values** check box to prevent SevOne NMS from using the settings it discovers from objects that are operationally down.
7. In the **Universal Collector** section:

a. In the **Days Without New Data Until Objects/Indicators Are Treated As Missing** field, enter the number of days that need to pass after the last data for an Object/Indicator before SevOne NMS starts treating it as missing during the routine Discovery. The minimum value that SevOne NMS allows is 1.

8. Click **Save** to save the Discovery settings.

11.1.3.5 Duration

The Duration subtab enables you to define how long to store data. You should consult with a SevOne Support Engineer before you change these settings to discuss the potential consequences of these changes.

1. In the **Device History Duration** field, enter the number of days to store Debug severity level device history. Info severity level history is stored for twice as long, Notice severity level history twice that, and so forth. The minimum value is 1 day and maximum value is 99 days. The default value is 7 days.

Example

Each log entry has an associated level (or severity) that follows the *syslog* standard. The higher the severity of the log, the longer it is kept. It follows the exponential model; low-level entries are trimmed.

Log Level	Duration
DEBUG	1 x <time> = 7 days
INFO	2 x <time> = 14 days
NOTICE	4 x <time> = 28 days
WARNING	8 x <time> = 56 days
ERROR	16 x <time> = 112 days
CRITICAL	32 x <time> = 224 days
ALERT	64 x <time> = 448 days
EMERGENCY	128 x <time> = 896 days

where, <time> is 7 days.

Optionally, device history duration can be set using the command line interface script.

Command Line Interface command

```
$ /usr/local/scripts/SevOne-act trim device [ARGUMENTS]


Arguments
--wait-duration         (Optional) This is how long to wait between cleaning up device logs
                        Default: 0
--server-id             (Optional) This is the peer id to run the script on
                        Default: 1
--short-term-logs-duration(Optional) This is how long to hold on to the short term logs, in
 seconds
                        Default: 604800
--emergency-purge       (Optional) Emergency purge. Force removing of old data with base 1 day
for debug logs.
```

In case of emergency, you may run the script with argument *--emergency-purge* which automatically sets device history duration to **1 day**. This results in 7x reduction for all device notes.

Command Line Interface command with --emergency-purge argument

```
$ /usr/local/scripts/SevOne-act trim device --emergency-purge
```

Log Level	Duration
DEBUG	1 x <time> = 1 days
INFO	2 x <time> = 2 days
NOTICE	4 x <time> = 4 days
WARNING	8 x <time> = 8 days
ERROR	16 x <time> = 16 days
CRITICAL	32 x <time> = 32 days
ALERT	64 x <time> = 64 days
EMERGENCY	128 x <time> = 128 days

where <time> is 1 day.

In comparison to Plugin Longterm data, Device Notes data is with low priority. The cron job is scheduled to run every day at 00:05 (GMT), an hour before Plugin Longterm Trim process execution.

```
5 0 * * * root /usr/local/scripts/SevOne-act trim device --log-timestamp --log-start-stop
2>&1 | logger -t SevOne-act-trim-device
```

If Plugin Longterm Trim is invoked manually with *--emergency-purge* argument, Device Notes Trim will first be called internally with the same arguments. In this case, more disk space will be reserved for the Plugin Longterm data. Only one instance of the process runs at a time.

2. In the **Logged Trap Duration** enter the number of days to store logged traps for display on the Logged Traps page. The minimum value is 1 day and maximum value is 365 days. The default value is 7 days.
3. In the **Unknown Trap Duration** field, enter the number of days to store unknown traps for display on the Unknown Traps page. The minimum value is 1 day and maximum value is 365 days. The default value is 1 day.
4. Click **Save** to save the Duration settings.

11.1.3.6 Email

The Email subtab enables you to define the email server that SevOne NMS uses to email reports and alerts. For details, please refer to sections **Report Properties** and **Alerts** in *SevOne NMS User Guide*.

The email server must be able to accept large attachments because a .pdf report can be over 20MB.

1. In the **Email Server** field, enter the host name or IP address of the SMTP email server for SevOne NMS to use to send emails.
2. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the email server.
3. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the email server.
4. In the **Email Sender** field, enter the email address to appear as the sender of the emails. This must have a valid email address format.
5. In the **Email Sender Name** field, enter the name to appear as the sender of the emails.
6. In the **Alerts Email Subject** field, enter the text to appear in the Subject line of alert emails. When you leave the Multiple Alerts Per Email check box clear, this field supports the variables listed below.
7. In the **Reports Email Subject** field, enter the text to appear in the Subject line of report emails. Supports the following variables: $name - Report name, $id - Report ID
8. Select the **Multiple Alerts Per Email** check box to place multiple alerts in the same email. Leave clear to receive each alert in a separate email. If you select this check box, the Alerts Email Subject does not include variables.
9. Select the **Email Cleared Alerts** check box to send an email when an alert clears.
10. Click the **Connection Security** drop-down and select a connection security protocol.
11. In the **Port** field, enter the port number on the email server for SevOne NMS to use.
12. Select the **Compress Emailed Reports** check box to compress the size of email attachments. Perform the following steps if you select this check box.
    a. In the **Compress Reports Larger Than** field, enter the minimum report size to compress. All smaller reports are not compressed. Enter 0 (zero) to compress all emailed reports.
    b. In the **Image Quality** field, enter how much to compress images (between 1 and 10). 10 = no compression, and the best quality and 1 = more compression and less quality. The default is 10.
13. Click **Send Test Email** to send a test email to the email address you associate to your user profile from the email sender through the email server.
14. Click **Save** to save the Email settings.

Alerts Email Subject Supported Variables

- $severity - Alert severity in text form
- $severityNum - Alert severity in numeric form
- $deviceId - ID of the alerting device
- $deviceIp - IP of the alerting device
- $deviceName - Name of the alerting device
- $deviceAltName - Alternate name of the alerting device
- $alertId - Alert identifier in numeric form
- $occurrences - Number of alert occurrences in numeric form
- $objectName - Name of the object that triggered the alert
- $objectAltName - Alternate name of the object that triggered the alert
- $thresholdId - Threshold identifier in numeric form
- $alertType - Type of the alert
- $threshold - Name of the threshold
- $policyId - Policy identifier in numeric format
- $policyName - Name of the policy
- $groupName - Device/Object Group name of the policy
- $message - Threshold trigger message
- $firstSeen - Time of the first alert.
- $lastSeen - Time of the last alert
- $assignedTo - Name of the user to which the alert is assigned
- $singleAlertMsg - Combination of severity and device name with format " - $severity: $deviceName"

11.1.3.7 Firewall

The Firewall subtab enables you to select the firewall service for the cluster. Click **Enable Cluster Firewall** check box to confirm and enable the firewall settings for the cluster. By default, it is disabled.

Click on **Open Port** to add a firewall port and click on **Remove Port** to remove user-added ports only.

11.1.3.8 FlowFalcon

The FlowFalcon subtab enables you to define how to collect and process raw flow data and aggregated flow data. An example at the end of this section sums up many of the following settings.

Changes to these settings can cause data loss. Please consult with your SevOne Support Engineer before you modify the FlowFalcon settings marked with an asterisk <*>.

1. * Select the **Store Raw Flow** check box to collect and store raw flow data. Most FlowFalcon views use raw data which provides more specificity in the result set at the tradeoff of longer report execution times and less historical data availability.
2. * Select the **Store Aggregated Flow** check box to collect and store the most relevant flow data in an aggregated format that aggregated FlowFalcon views use for faster report execution times.
3. * In the **Raw Flow Duration** field, enter the number of days' worth of raw flow data to keep. Gigabytes of raw flow data can accumulate quickly. You define aggregated flow duration on the Cluster Manager at the peer level as described later in this topic. The minimum value is 0 days. The default value is 1 day.
4. * In the **Raw Flow Data Size** field, enter the maximum amount of disk space to allocate for raw flow data. The minimum value is 0 GB. The default value is 100 GB.
5. * In the **Write Interval** field, enter the number of seconds to collect flow data before creating a flat file and writing the data to the disk (between 60 and 300). The default is 60, which is recommended. A longer write interval results in fewer (but larger) flat files for raw data and smaller tables for aggregated data. See example below.
6. Select the **Drop Long Flows** check box and enter the maximum number of seconds to consider flow data "long" in the **Max Flow Duration** field (between 60 and 600). The default is 120. This drops flows when the flow's duration exceeds the write interval. Long flows are usually due to improper router configuration. This setting triggers an administrative message that appears upon log on to inform you to review the router configuration. Suggested Max Flow Duration is ~2x the Write Interval from the previous step.
7. Select the **Enable ASN/Country Enrichment** check box to enable enrichment of flow with ASN (Autonomous System Number) and Country determined from the IP addresses in the flow. When enabled, flow is matched as it arrives to a country and ASN in the table; the ASN and Country information is stored along with the flow. Available for both raw and aggregated flow. By default, this field is enabled. Enriched views are enabled by default but, only apply to *raw* flow data. At present, there is a limit of 10 aggregated views your appliance can support. Due to this limit, the views are delivered as *raw* data. However, you can aggregate as needed. Please refer to FlowFalcon View Editor for details.
8. Select the **Enable Service Enrichment** check box to enable service enrichment for flow collection and reporting. When enabled, flow is matched as it arrives to a service profile in the service profile table and the Service Profile Id is stored along with the flow. Available for both raw and aggregated flow. By default, this field is enabled. At present, there is a limit of 10 aggregated views your appliance can support. Due to this limit, the views are delivered as *raw* data. However, you can aggregate as needed. Please refer to *FlowFalcon View Editor* for details.

   The **Service Profiles** can be found in **Administration** > **Flow Configuration** > **Protocols and Services** > **Service Mapping** tab.
9. Select the **Enable MPLS Attribute Mapping** check box and enter the number of seconds for how frequently to read the map files and to refresh the mapping in the **MPLS Attribute Mapping Refresh Interval** field. This enables you to map v9 NetFlow template data from core "P" routers for reports that use the following fields in FlowFalcon views: 45050: Customer Client IP, 45051: Customer Client Subnet, 45052: Customer VRF Name, 45053: Customer Application IP, 45054: Customer Application Subnet, 45055: PE Ingress IP, and 45056: PE Egress IP.

   Map files are customer-specific. The MPLS Flow Mapping page enables you to upload the two required map files into SevOne NMS.
10. * In the **Aggregation TopN** field, enter the number of results (between 50 and 1000) to store for each aggregation per each write interval. This consumes disk space and is the maximum number of individual results that an aggregated FlowFalcon view can display. The default value is 100.

    Warning: Setting a value greater than the default may result in data loss.
11. In the **Hide Inactive** field, enter the number of days (minimum 1) to display data for an inactive device or interface before the device or interface is considered inactive and its information is hidden. The default is 14. A device or interface is considered inactive if it does not send data to SevOne NMS.
12. In the **Deny Inactive** field, enter the number of days (minimum 0) to deny an interface that is inactive (does not send data) for this many days. The default value is 0 days; i.e., disabled. If an interface is found to have no data for the defined number of days, the process denies the interface in the Flow Interface Manager. When all interfaces for a device are denied, the device is also denied. Upon denial, licenses / objects that were in use are freed up for the denied interface(s).
13. In the **Purge Inactive** field, enter the number of days (minimum 0) to store data for an inactive device or interface. The default is 0. Enter 0 (zero) to never purge data.
14. In the **Incoming Port** field, enter the port number on the SevOne appliance to listen for flow traffic.

15. Click the **Raw Data Compression** drop-down and select a method for compressing raw data files. Greater compression requires less storage but results in higher CPU usage.
16. Select the **Display Flow Sample Rates** check box to display the sampled flow rate on FlowFalcon reports that contain split interfaces and to display an additional column on the Flow Interface Manager for sampled data. FlowFalcon reports with sampled data display a message. Interfaces that are not sampled use a sample rate of 1X.
17. Select the **Create Egress Records When Not Available** check box to automatically create egress records for ingress interfaces that do not receive egress records. Leave clear if your devices support both ingress and egress interface flow export. This does not affect how SevOne NMS handles NetFlow v5.
18. Select the **Create Ingress Records When Not Available** check box to automatically create ingress records for egress interfaces that do not receive ingress records. Leave clear if your devices support both ingress and egress interface flow export. This does not affect how SevOne NMS handles NetFlow v5.
19. Select the **NAT Support** check box to enable support for routers behind network address translation (NAT).
20. In the **Max Write Threads** field, enter the maximum write threads for Flow Traffic. The minimum value is 1 thread and the maximum value is 10 threads. The default value is 1 thread.
21. Click **Save** to save the FlowFalcon settings.

Example

This example uses flows that come from a single device/interface/direction to compare raw and aggregated data at both ends of the settings spectrum (60 to 300 seconds) when flows are received at a rate of 100 flows/minute and each flow is 50 bytes.
Raw - All flows collected during each write interval are written to the disk in a single file. A longer write interval results in larger file sizes, but fewer files (since they are written less often). For a flow rate of 100 flows/minute at 50 bytes each over a 10 minute time frame.

- 60 second write interval: 10 files are written, one file per minute. Each file contains 100 flows resulting in 5000 bytes per file. (10 x 5 KB files = More smaller files)
- 300 second write interval: 2 files are written, one file every 5 minutes. Each file contains 500 flows resulting in 25,000 bytes per file. (2 x 25 KB files = Fewer larger files).

Both approaches result in the same amount of disk usage (in this case 50 KB).
Aggregated - At the end of each write interval, SevOne NMS calculates one data point each for the number of results you enter as the Aggregated TopN per aggregated view and writes those <n> data points to the database (default - 100). Using a 10 minute time span:

- 60 second write interval: Writes 100 data points every minute and adds a total of 1000 records to the database.
- 300 second write interval: Writes 100 data points every 5 minutes and adds a total of 200 records to the database.

Thus a larger write interval results in fewer entries to the database and is why a longer time period results in smaller tables.
For every write interval (in this case 60 seconds), SevOne NMS determines the top <n> for every device, interface, direction, aggregated view combination (e.g., Router 1, Eth0/0, Incoming would provide the top 100 data points for every aggregated view (Top Talkers, Top Conversations, etc.). Then SevOne NMS determines a top 100 for Router 1, Eth0/0, Outgoing for every aggregated view. This process continues for each Interface on every device.
All flows that do not make it into the top 100 are aggregated together into a single record called *Remaining Traffic*. This happens for every device, interface, direction, view combination. Total Traffic is the top <n> plus remaining traffic to represent all traffic in the network.

11.1.3.9 FTP

The FTP subtab enables you to define the FTP destination settings for SevOne NMS to use when you send a report via FTP. For details, please refer to section **Report Properties** in *SevOne NMS User Guide*.

1. In the **Server** field, enter the IP address or host name of the FTP server where SevOne NMS is to send reports.
2. In the **Port** field, enter the port to which SevOne NMS is to send reports.
3. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the FTP server.
4. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the FTP server.
5. In the **Path** field, enter the path to the location on the FTP server where you want the report to be sent.
6. Click **Test FTP Settings** to verify that your FTP settings work correctly.
7. Click **Save** to save the FTP settings.

11.1.3.10 General

The General subtab enables you to define general system settings.

1. In the **Cluster Name** field, enter the name of your SevOne NMS cluster. The name entered will appear in your web browser tab if **Cluster Settings** tab > **Alerts** subtab > **Show Alerts in Title** field is disabled.
2. Click the **Log Entry Severity Level** drop-down and select the severity level at which to write to the log file. Select a lower severity level to generate more log data. This setting is primarily for use by SevOne Support Engineers.
3. Click the **Search Behavior** drop-down and select one of the following options:
   - **Default**. Searches with special characters are exact searches. Searches without special characters are wildcard searches.
   - **Never Exact**. All searches are wildcard searches.
   - **Always Exact**. All searches are exact searches.
4. Select the **Override Precision** check box to specify the level of precision. Data typically rounds to two decimal places. Select this check box and enter the number of decimal places to which to round data (between 0 and 6). The default is 0. Most report workflows enable you to define the precision for each report.
5. In the **Peer Status Cache** field, specify the number of seconds between updates to the peer availability cache. SevOne advises against entering 0 or high values (anything over 60 seconds) without first contacting SevOne. (A cache invalidation setting of 0 will not rebuild the cache.) The minimum value is 0 seconds and the maximum value is 999999 seconds.
6. Select the **API Weekend Work Hours** check box if you use the SevOne NMS API and your work hours include weekends.
7. Select the **Measure System Uptime** check box to populate the Deferred Data plugin for each device with a system object and a SysUpTime indicator that contains the normalized data. The deferred data SysUpTime is the true representation of the devices uptime as SevOne NMS derives from polls every 15 minutes. Each poll collects data for the past 7 days. System uptime is the length of time a device has been up without any downtime (loss of connection to the device can appear as downtime). For details, please refer to section **Deferred Data Plugin** in *SevOne NMS User Guide*.
8. Select the **Reports Restricted By Default** check box to restrict access to new reports. You can override this setting for each report on the Report Properties. For details, please refer to section **Report Properties** in *SevOne NMS User Guide*.
9. Select the **Use WebKit to PDF** check box to use WebKit to render .pdf reports. WebKit integration is a beta feature but has proven to generate more presentable .pdf's. **NOTE**: If a Performance Metrics Report is added with more than 25 indicators (which are in single-row table) along with the graph, the report will shrink-to-fit on the page.
10. Select the **Fit PM Graph in One Page** check box to shrink the Performance Metrics graph to fit on a single page while using Webkit to render .pdf reports.
11. Select the **Enable Localization** check box to enable the display of SevOne NMS in a language other than English. When you select this check box, click the **Default Language** drop-down and select the language to appear by default in user definition workflows and to display on the Login page. Localization is a beta feature and can be set for each user on the Preferences page. For details, please refer to section **Preferences** in *SevOne NMS User Guide*.
12. Click the **Week starts on** drop-down and select the start day of the week. Options are Saturday, Sunday, or Monday. This can also be set on the Preferences page.

    The value set in **Week starts on** field here overwrites any user specified setting when reports are mailed. For example,
    Cluster Manager > **Cluster Settings** > **General** > **Week starts on** is set to *Sunday*.
    Administration > **My Preferences** > **Week starts on** is set to *Monday*.
    When reports are mailed, it will choose the day set in Cluster Manager (Sunday as shown in this example) as the week's start day.

13. In the **Peer Takeover Threshold** field, specify the number of minutes for the self-monitoring notification to be pushed when the peer takeover exceeds <n> minutes to complete. The default value is 10 minutes. The minimum value is 1 minute and maximum value is 999 minutes.
14. Select the **Show Hostname in Title** check box to add the hostname to the web browser tab.
15. Select the **Enable Admin Notifications** check box to enable email receipt of Admin Notifications.

    If the Alertmanager service is stopped and then, email configuration is modified, the updates to the configuration will only take effect after the Alertmanager service is restarted.

    Stop Alertmanager

    ```
    $ supervisorctl stop alertmanager
    ```

    Restart Alertmanager

    ```
    $ supervisorctl restart alertmanager
    ```

16. In the **Cluster Time Drift Threshold** field, specify the number of seconds allowed in time drift. The admin will be notified when an appliance exceeds the configured threshold for time drift as compared to the the system time on the Cluster Leader. The default value is 60 seconds. The minimum value is 15 seconds and maximum value is 300 seconds.
17. Select the **Alert NTP server unavailable** check box to notify if there are no **active** NTP servers.
18. Click the **IPv6 address representation** drop-down and select one of the following options:
    - **Full Format** - to display IPv6 address in full format. For example, 1080:0000:0000:2601:0000:0800:200c:417a
    - **Zero Compression with Drop Zero** - replaces the consecutive blocks of zeros with a double colon for the first contiguous block only. Drops/omits the leading zeros and not the trailing zeros of the rest where applicable. For example: 1080::2601:0:0800:200c:417a
    - **Drop Zero** - drops/omits the leading zeros and not the trailing zeros. For example, 1080:0:0:2601:0:800:200c:417a
    - **Zero Compression** - replaces the consecutive blocks of zeros with a double colon for the first contiguous block only. For example: 1080::2601:0000:0800:200c:417a
19. Select the **Disk Emergency Mode** check box to enable or disable the disk emergency mode safety checks.

    - On an active appliance, script **disk-emergency-mode** runs on a cron job every 15 minutes to determine whether the peer needs to be put into disk emergency mode.
    - Field **Disk Emergency Mode** must be **enabled** if all of the following conditions are met.
        - disk use exceeds **max_disk_util**.
        - you have enabled this feature and configured a threshold (for example, 50%) for **/data** utilization by **non-longterm-data**.
        - a ratio greater than this threshold is being used by **non-longterm-data** in **/data**. For example, 80% of /data is filled with logs; a critical error state. This implies a situation that cannot be recovered from by automated means and requires **admin** intervention.
    - When field **Disk Emergency Mode** is enabled,
        - an *admin* message is presented to you at login.
        - if prometheus is enabled, a prometheus alert is generated.
        - in **SevOne-act trim longterm**, no **disk-use-based** trim will take place. Duration-based trim will continue as normal.
        - **updater** does not commit any *polling shortterm* data to risk.
        - **SevOne-ffupdater** does not commit any *flow shortterm* data to risk.
        - **SevOne-trapd** does not write traps to disk.
        - **SevOne-netflowd** does not collect raw data.
        - polling and collection continue as normal; the downstream services such as SevOne Data Bus, Dataminer, etc. operate as expected with **shortterm** data as long as **mysqld** has not stopped completely due to the disk (/data) being 100% full.
    - If the conditions required for Disk Emergency Mode are remediated, the mode is turned off.
    - This mode runs on all PAS peers in a cluster including *passive* appliances.
    - The status of Disk Emergency Mode is stored in **/SevOne/appliance/settings/disk_emergency_mode** comprised of **1** or **0** / **on** or **off**.
    - Disk emergency script is logged in **/var/SevOne/SevOne-disk-emergency-mode.log** file.

20. In the **Disk Emergency Threshold** field, specify the threshold percent of non-MySQL data allowed to occupy /data. The minimum percent is 20% and maximum percent is 90%. The default value is 90%.
21. Click **Save** to save the General settings.

11.1.3.11 Graphs

The Graphs subtab enables you to define the settings for the graphs that appear in reports. For details, please refer to section **Report Manager** in *SevOne NMS User Guide*.

1. Select the **Abbreviate Graph Text** check box to abbreviate long names on the graph with ellipses.
2. Select the **Display Poll Frequency** check box to have the Display Frequency check box selected by default in report creation workflows.
3. Select the **Display Minimum Value** check box to have the Display Minimum check box selected by default in report creation workflows.
4. Select the **Draw Horizontal Grid Lines** check box to display horizontal lines on the graph then enter how close the horizontal lines should be to one another in the **Horizontal Grid Density** field.
5. Select the **Draw Vertical Grid Lines** check box to display vertical lines on the graph enter how close the vertical lines should be to one another in the **Vertical Grid Density** field.
6. Select the **Display Last Poll Value** check box to display the value of the last successful poll in the graph legend.
7. Select the **Display Units in TopN CSV** check box to append the units as an additional column in the TopN CSV exports.

8. Click the **Default Aggregation Alignment** drop-down and select **Aligned to Interval** or **Aligned to Start Time** to allow you to align all the aggregation points by Interval or by Start Time.
9. Select the **De-normalizing GAUGE Totals** check box to perform a total aggregation instead of using a simple sum for the **Total** column in graph summaries.
10. Click **Save** to save the Graphs settings.

11.1.3.12 ICMP

The ICMP subtab enables you to define ICMP settings for devices on which you enable the ICMP plugin. For details, please refer to section **ICMP Plugin** in *SevOne NMS User Guide*.

1. Select the **Always 100% Availability** check box to report 100% availability in ICMP even if a single packet makes it through. Leave clear to set availability to the percentage of the packets that make it through.
2. Click **Save** to save the ICMP settings.

11.1.3.13 IP SLA

The IP SLA subtab enables you to define IP SLA settings for the devices on which you enable the IP SLA plugin. You can override this setting for individual devices from the Edit Device page. For details, please refer to sections **IP SLA Plugin** and **Edit Device** in *SevOne NMS User Guide*.

1. Click the **Default Responder Action** drop-down.
    - Select **Ignore** to have SevOne NMS not change the IP SLA responder setting on devices.
    - Select **Yes** to turn on the IP SLA responder on devices upon discovery, when possible.
    - Select **No** to turn off the IP SLA responder on devices upon discovery, when possible.
2. Click **Save** to save the IP SLA settings.

11.1.3.14 Logging

The Logging subtab enables you to manage which user actions are to create log entries. You can view log entries on the Cluster Manager at the appliance level on the System Logs tab. See the Processes and Logs topic for a list of the system logs to where log entries are made.

-  - User actions are logged.
-  - User actions are not logged.

You can override cluster level Logging settings at the peer level from the **Peer Settings** tab described later in this topic. Some user action log functionality is dependent upon your software kernel version being higher than 2.6.36. On the **Administration** > **About** page, click **PHP Status** under **Status Information** to find your kernel version.

- applianceSettingManaged - Cluster Manager Appliance Settings creates log entries when a user changes a setting on the Cluster Manager Appliance Settings tab.
- clusterManaged - Cluster Manager Appliance Management creates log entries when a user performs actions such as database synchronization, fail over, etc. from the gear menu at the appliance level on the Cluster Manager.
- commandExecuted - Console Command Execution creates log entries when a user executes a command in the Linux terminal.
- configFileModified - System Configuration Files creates log entries when various system configuration files are modified.
- devicePluginEntityManaged - Device Editor Plugin Object Managers creates log entries when a device plugin object manager (e.g. "DNS Objects", "ICMP Objects" or "HTTP Objects") is modified.
- devicePluginManaged - Device Editor Plugin Settings creates log entries when a user modifies the plugin settings for a device on the Add/Edit Device page.
- discoveryManaged - Discovery Management creates log entries when a user queues a device discovery, changes discovery priority or cancels discovery.
- entityManaged - General Management triggers when a user creates, updates, deletes, enables or disables devices, alerts, thresholds, policies, users, trap destinations, and others.
- entityMappingManaged - Association Management creates log entries when a user modifies associations of device/object groups, nested device/object groups, user roles, trap destinations or metadata mapping.
- fileUploaded - File Upload Management creates log entries when a file has been uploaded to cluster manager upload update file, status maps or device types.
- importTriggered - Data Import creates log entries when a user imports data via an .spk file.
- processManaged - Cluster Manager Processes creates log entries when a user starts, stops, or restarts a process from the Process Overview tab on the Cluster Manager.
- ruleApplied - Membership Rules triggers when a user applies object group and device group membership rules.

- settingModified - Cluster Manager Settings creates log entries when a user modifies the settings on the **Cluster Settings** tab or the **Peer Settings** tab on the Cluster Manager.
- soapMethodInvoked - SOAP API Call creates log entries when a user invokes a SOAP API call.
- userAuth - User Authentication creates log entries when a user logs in, logs out or is affected by other authentication events such as inactivity time out or failed login attempts.
- userPasswordChanged - User Password creates log entries when a user changes their password or an account is created with a new password.

Click **Save** to save the Logging settings.

11.1.3.15 Login

The Login subtab enables you to add a custom message to the Login page and to display the Alert Summary, Instant Status, and Alerts on the Welcome Dashboard.

1. In the **Login Page Message** field, enter the message to appear on the Login page. Limit is 1500 characters and you cannot use HTML formatting.
2. Select the **Use a Fixed Width Font** check box to use a font whose letters and characters each occupy the same amount of horizontal space. The font you define for your browser is the default font for the message. To change your font in Internet Explorer; click the Tools menu and select Internet Options then Fonts. To change your font in Firefox: click the Tools menu, select Options, and then select Content.
3. In the **Welcome Dashboard** section:
    a. Select the **Display Alert Summary** check box to display an Alert Summary report on the Welcome Dashboard. For details, please refer to section **Alert Summary** in *SevOne NMS User Guide*.
    b. Select the **Display Instant Status** check box to display an Instant Status report on the Welcome Dashboard. For details, please refer to section **Instant Status** in *SevOne NMS User Guide*.
    c. Select the **Display Alerts** check box to display an Alerts section on the Welcome Dashboard. For details, please refer to section **Alerts** in *SevOne NMS User Guide*.
4. In the **User Sessions** section, select the **Allow Concurrent User Sessions** check box to allow a user to login in more than once, concurrently, using the same login credentials.
5. In the **Single Sign-On** section,
    a. Select the **Enable Single Sign-On** check box to allow a user to use the configured Single Sign-On integrations instead of the default authentication.
    b. Select the **Enable Peer Certificate Verification** check box to verify the peer's certificate when logging in with Single Sign-On.
    c. In the **OpenID-Connect Issuer URL** field, enter the issuer URL to use for Single Sign-On integrations.

    > The **OpenID-Connect Issuer URL** must match the **Nginx Server Certificate Common Name**. For example, if the server certificate common name is **sso.example.com**, the OpenID-Connect Issuer URL must be **https://sso.example.com/sso**.

    d. In the **OpenID-Connect Client ID** field, enter the string to identify SevOne NMS for Single Sign-On integrations.
    e. In the **OpenID-Connect Client Secret** field, enter the secret to identify SevOne NMS for Single Sign-On integrations.

    > **Root certificate (and chain) must be added to the operating system trust store**
    >
    > **Example**
    >
    > ```
    > $ SevOne-act add-certificates --file /path/to/cert.crt
    > ```

6. Click **Save** to save the Login settings.

11.1.3.16 Poller

The Poller subtab enables you to define poller settings.

1. In the **Poller Threads** field, enter the number of poller threads to use concurrently (between 1 and 1000). The default is 60.

SevOne NMS Appliance Model	Recommended Poller Thread Max
PAS2k	60
PAS5k	60
PAS10k	100
PAS20k	200
PAS60k	300
PAS100k	600
PAS200k	1000

> ⚠ Poller Thread Max should be set to the smallest size of the SevOne NMS appliance model in the cluster. By not doing so, it may result in resource issues.

2. In the **Update Interval** field, enter the number of seconds to collect poll data before writing data to the disk (between 1 and 300). The default is 60.
3. Select the **Display Poller Downtime** check box to display a gap in a graph when a poller is down. This field is strictly related to the data that is polled via **SevOne-polld** and does not apply to any external data pushed to the system. In the **Poller Downtime Threshold** field, enter the number of seconds for polling to be down before a gap appears in a graph. Leave clear to display a continuous line in graphs between actual poll points. The minimum value for Poller Downtime Threshold field is 1 second.
4. In the **Oracle Plugin Timeout** field, enter the number of minutes to keep each Oracle poller thread polling before time out and start over. This prevents threads from being consumed by infinite Oracle poll times. The minimum value is 1 second and the maximum value is 32767 seconds.
5. In the **SNMP Timeout** field, enter the number of seconds until timeout. The minimum value that SevOne NMS allows is 1 second. The default is 3 seconds.
6. In the **SNMP Retries** field, enter the number of retries before SNMP gives up on the timeout. The default is 3.
7. Select the **Enable Custom Calculation Poller Cutoff Period** check box to enable custom calculation poller cutoff period for all devices in the cluster. In the **Calculation Poller Custom Cutoff Period**, enter the number of minutes that calculation poller will treat data buckets as valid. Calculation Poller Custom Cutoff Period field can range between 5 minutes to 120 minutes (2 hours). The default value is 5 minutes.

   This field allows an administrator to define a duration between this range. The Calculation Poller looks for data for the indicators if the current values are not available. By updating this field, it affects all Calculation Poller devices on the next poll. For example, if a calculation object is comprised of two SNMP indicators that are normally polled every 5 minutes and, one of those indicators is no longer available, then the Calculation Poller uses the last available value for that indicator. This will prevent calculation objects from returning *null* values but may result in calculated values that do not accurately reflect the raw data.

   When a customized cutoff period is not used for Calculation poller, **SevOne-polld** dynamically assigns an availability cutoff period of *x2 the device's polling frequency*.

   > ⚠ Since the default polling period is 5 minutes, devices that derive data from non-polled sources such as, xStats, may require the use of a custom cutoff period to avoid null calculation values when current data is not available.

8. If **SevOne-polld** fails to poll the device, an *admin* may select the criteria required for a device to be considered unavailable. Under **Device Unavailability by Plugin Type**, configure the following fields.
   a. Select **Criteria for Device Unavailability** drop-down and choose the desired criteria.
      - **All** - Marks devices as unavailable if **all** of the checked plugins are found unavailable.
      - **Any** - Marks devices as unavailable if **any** of the checked plugins are found unavailable.
   b. Select the **SNMP** check box to select the SNMP plugin.

c. Select the **ICMP** check box to select the ICMP plugin.

9. Click **Save** to save the Poller settings.

11.1.3.17 Ports

The Ports subtab enables you to define the port settings for communication between peers in the cluster.

1. The **Primary/Secondary Port** field displays the TCP port for communication between the Primary appliance and the Secondary appliance in a Hot Standby Appliance peer pair. Do not change. This port is for internal use.
2. The **Alert Server Port** field displays the port for alerts. Do not change. This port is for internal use.
3. In the **Trap Receiver Port** field, enter the UDP port number on the SevOne appliance to listen for incoming SNMP traps.
4. The **SevOne-gui-installer Port** field displays the TCP port number required for SevOne-gui-installer. The default value is **9443**. You may change the port number to any other valid value. The port will get opened automatically if firewall is enabled on the system. After changing the port, you must go to **Cluster Manager** > **Cluster Upgrade** and click on **Run Installer** to generate the new URL.
5. Click **Save** to save the Ports settings.

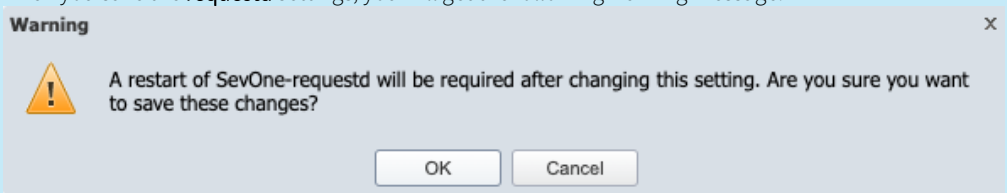
11.1.3.18 Requestd

The Requestd subtab allows you to configure **SevOne-requestd** runtime parameters for the cluster.

1. The **Responder Queue Size** field allows you to set the number of responder tasks to queue up. Maximum number of queries from remote peers that queue up for the local peers to reply to, as the responder threads become available. The default value is 400. The queue size can range between 400 and 1200.
2. The **Local Threads** field allows you to set the maximum number of worker threads used for internal **requestd** requests made to the local appliance. The default value is 200. The threads can range between 200 and 600.
3. The **Originator Threads** field allows you to set the maximum number of worker threads from the originator. These threads are used for executing the requests from the local appliance (the originator) to the remote appliances. The originator threads are requests that distribute tasks to other peers. The default value is 200. The threads can range between 200 and 600.
4. The **Responder Threads** field allows you to set the maximum number of threads used for responding to remote requests from other appliances. The default value is 200. The threads can range between 200 and 600.
5. The **Requestd Module Originator ZMQ Timeout** field allows you to set the timeout for the originator ZMQ process that handles the **requestd** queries. 0 minutes indicates no timeout. Timed out queries are discarded, resulting in query failure. Lowering the timeout may help **requestd** from exhausting threads due to excessively long queries or network conditions that may cause ZMQ to wait indefinitely. Setting the value too low may cause reports that are expected to take a long time to run, to timeout or display impartial results. The valid values are 0 minutes (for **no timeout**) or 15 - 1440 minutes. The default value is 0 minutes.
6. Click **Save** to save the Requestd settings.

> **IMPORTANT**
>
> These settings are provided to support advanced NMS troubleshooting. NMS administrators are strongly discouraged from making changes to these settings without first contacting SevOne Support. Improper changes to these settings may cause service degradation or disruption.

When you save the **requestd** settings, you will get the following warning message.



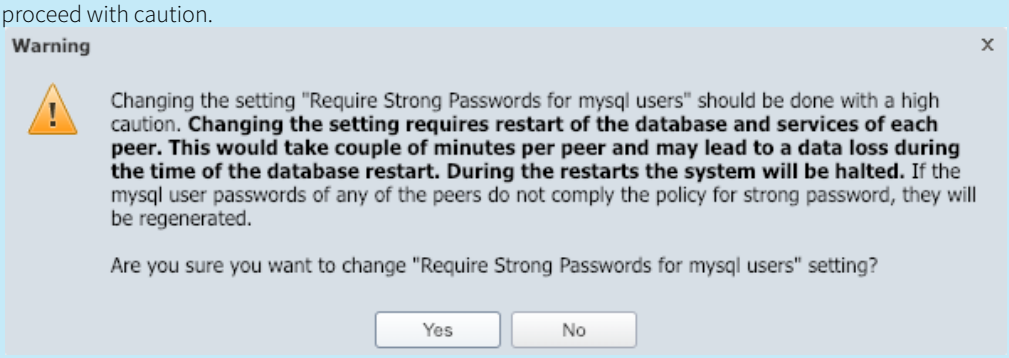
Click **OK** to save the settings or Cancel to exit.

11.1.3.19 Security

The Security subtab enables you to define security settings.

1. In the **Inactivity Timeout** field, enter the number of minutes a user can remain inactive before SevOne NMS automatically logs the user out of the application (between 5 and 86400). The default is 30. You can override this setting for each individual user from the User Manager.
2. Select the **Enable Hard Timeout** check box to enable hard timeout for all users in the cluster with the exception of the **admin** user. Enable the check box to allow you to enter the number of minutes in **Hard Timeout** field the user can remain alive before SevOne NMS automatically logs them out of the application. The default value is 30 minutes. Hard Timeout field can range between 5 minute to 86400 minutes (60 days).
3. In the **Minimum Password Length** field, enter the number of characters users must have in their password (between 0 and 99). The default is 0. Enter 0 (zero) to disable this feature.
4. In the **Enforce Password History** field, enter the number of password changes a user must make before they can repeat a password (between 0 and 999). The default is 0.
5. In the **Minimum Password Age** field enter the number of days a user must wait between password changes (between 0 and 999). The default is 0. This feature prevents users from circumventing Password History enforcement. Enter 0 (zero) to disable this feature.
6. In the **Password Change Notification** field, enter the number of days to wait after a password change before a user receives a password change notification (between 0 and 999). The default is 0. Enter 0 (zero) to disable this feature.
7. In the **Maximum Password Age** field, enter the number of days a user account can remain enabled before the user must change their password (between 0 and 999). The default is 0. Enter 0 (zero) to disable this feature.
8. Select the **Mask Read Community String** check box to mask Read Community Strings on user interfaces. Write Community Strings are masked by default.
9. Select the **Require Strong Passwords** check box to enforce the complexity of user passwords. If you select this check box, passwords must contain at least one special character **!@#$%^&*=+_?<>/~()-[{]}|\;:",** and at least two of the following three types of characters: lowercase letters, UPPERCASE letters, and numbers. In addition, passwords cannot contain more than two of a given type of character in succession (upper and lowercase letters count as the same type). An example of a valid password: 8s0h43o@7!o&p3. If your current password does not meet this requirement, you will be forced to change the password at the next log on.
10. Select the **Require Strong Passwords for mysql users** check box to enforce the complexity of MySQL user passwords. If you select this check box, minimum length of the MySQL password must be at least 14 symbols long, contain at least one special character **+-_@[]:,.%**, at least one number, at least one UPPERCASE letter, and at least one lowercase letter. The valid characters are **a-z, A-Z, 0-9, +-_@[]:,.%**. The invalid characters are ***$!#^;&**. An example of a valid password: **8s0H43o@7]o%p3**. Current MySQL passwords that do not meet this requirement will be changed to a random, compliant password.

By selecting the check box, the following warning message will appear. Please read the warning message and proceed with caution.



Select **No** to make no changes.
Select **Yes** will require a restart of the MySQL database and services of each peer. Click on **Save** button to apply the changes. This action may result in you intermittently losing access to SevOne NMS User Interface as the MySQL services are being restarted. Depending on the cluster and load per appliance, times will vary.
Log in to the SevOne NMS via the Command Line Interface (CLI) as user **root** to check the progress of the restart of **mysqld** services. Execute the command below.

Check mysqld services restart progress

```
$ SevOne-peer-do "supervisorctl status mysqld mysqld2"
```

Example: View status of MySQL services

```
$ SevOne-peer-do "supervisorctl status mysqld mysqld2"
--- Gathering peer list...
--- Running command on 10.168.116.40...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 14358, uptime 0:00:12
mysqld2 STARTING
Connection to 10.168.116.40 closed.
[ OKAY ]
--- Running command on 10.168.117.67...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 5043, uptime 1:21:47
mysqld2 RUNNING pid 5085, uptime 1:21:36
Connection to 10.168.117.67 closed.
[ OKAY ]
--- Running command on 10.168.118.17...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 17089, uptime 1:20:49
mysqld2 RUNNING pid 17206, uptime 1:20:35
Connection to 10.168.118.17 closed.
[ OKAY ]
--- Running command on 10.168.117.30...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 16234, uptime 1:19:51
mysqld2 RUNNING pid 16355, uptime 1:19:36
Connection to 10.168.117.30 closed.
[ OKAY ]
```

⚠ The processes are seen in transition between **RUNNING** and **STARTING** but you have to wait until all the peers have the **mysqld** and **mysqld2** services in **RUNNING** state and the **uptime** is seen as close to the current time. The ones highlighted in red have still not restarted and are yet to be processed - the process is performed one peer at a time.

Example: Completion of MySQL services after restart on all appliances

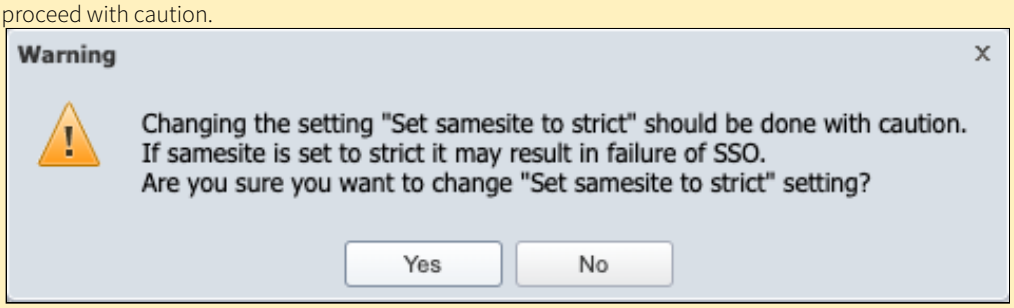
```
$ SevOne-peer-do "supervisorctl status mysqld mysqld2"
--- Gathering peer list...
--- Running command on 10.168.116.40...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 14731, uptime 0:04:35
mysqld2 RUNNING pid 14873, uptime 0:04:13
Connection to 10.168.116.40 closed.
[ OKAY ]
--- Running command on 10.168.117.67...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 22565, uptime 0:02:58
mysqld2 RUNNING pid 22800, uptime 0:02:43
Connection to 10.168.117.67 closed.
[ OKAY ]
--- Running command on 10.168.118.17...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 9207, uptime 0:01:54
mysqld2 RUNNING pid 9267, uptime 0:01:33
Connection to 10.168.118.17 closed.
[ OKAY ]
```

```
--- Running command on 10.168.117.30...
Authorized uses only. All activity may be monitored and reported.
mysqld RUNNING pid 7949, uptime 0:00:38
mysqld2 RUNNING pid 7996, uptime 0:00:23
Connection to 10.168.117.30 closed.
[ OKAY ]
```

> This indicates that the setting for **Require Strong Passwords for mysql users** is now enabled. If the password was not already set for the MySQL user, or if the existing password did not meet the secure complexity requirements, then SevOne NMS will automatically set a password that meets these requirements, but the actual password may not be known to the user. In such cases, you can optionally change the password and meet the complexity requirements. Please refer to *SevOne Data Platform Security Guide*, section **Change MySQL User Credentials** for details on how to change the MySQL user credentials.

11. Select the **Allow Forcelogin** check box to enable SevOne NMS integration with other software applications via the Forcelogin script.
12. Select the **Force Same Origin Policy** check box to prevent SevOne NMS from being loaded outside of the current domain. This includes portals and the use of the force login script to load SevOne NMS into an iframe from where a malicious user could log a user's activity. **Note:** If you clear this check box, the application security is lowered in a way that can prevent SevOne NMS from passing specific security scans.
13. Select the **Rest API Validate Certificates** check box to enforce REST API to validate the certificates of the other appliances when calling their REST API services.
14. Select the **Require HTTPS** check box to require a secure connection for all dynamic content. You must log on via HTTPS to enable this check box.
15. Select the **Allow insecure code in Simple attachment** check box for SevOne NMS administrators to allow/disallow usage of custom code in the Simple attachments.
16. **Enable render graph security** option, when unchecked, allows the administrator to share the exact URL (for example, *http://<SevOne NMS IP address>/doms/graphs/renderGraph.php?is[]=1%3A7983%3A471642%3A834&timespan=Today*) of the report with a user who has more restrictive (reduced) permissions. The user can enter the exact URL in the browser to view the entire contents of the report. However, for security reasons, it is highly recommended that this option is always checked to prevent users with more restrictive (reduced) permissions from crafting the URL to view a report.
17. Select **Set samesite to strict** check box to set SameSite cookie, **session.cookie_samesite**, to **strict**. By default, the check box is unchecked. i.e., SameSite cookie, **session.cookie_samesite**, is set to **lax**.

> By selecting the check box, the following warning message will appear. Please read the warning message and proceed with caution.



18. In the **Account Lockout** section:
    a. In the **Disable Inactive Users** field, enter the number of days a user can go without logging on before their account is disabled (between 0 and 999). The default is 0. Enter 0 (zero) to disable this feature, so that inactive users will never be disabled.

    > **Note:** This setting does not affect the Guest users you define on the Authentication Settings page for LDAP, TACACS, and RADIUS; nor does it affect the "admin" user.

    b. In the **Threshold** field, enter the number of incorrect log on attempts a user can make (within the Counter Reset time span) before the account is locked. Enter 0 (zero) to disable this feature. **Note:** When you set this to anything other than 0 (zero), log on becomes dependent upon validation from the cluster leader peer. If the cluster leader peer is

not accessible from a peer on which a user attempts to log on, access to the application will not be available. The minimum value is 0 attempts and the maximum value is 99999 attempts.

c. If you enter a number in the Threshold field, in the **Counter Reset** field, enter the number of minutes during which the user enters an incorrect user name and password combination before the account is locked. Set this to 0 (zero) to disable this feature. **Example:** Enter 3 as the Threshold and 2 as the Counter Reset. If the user incorrectly enters their user name and password combination three times in a two minute time span, the account is locked for the number of minutes you enter in the Duration field. The minimum value is 0 minutes and the maximum value is 99999 minutes.

d. If you enter a number in the Threshold field, in the **Duration** field, enter the number of minutes for the account to be locked after the Threshold/Counter Reset combination is exceeded (between 0 - minimum value and 99999 - maximum value). The default is 0.

19. Click **Save** to save the Security settings.

11.1.3.20 SFTP

The SFTP subtab enables you to define the SFTP destination settings for SevOne NMS to use when you send a report via SFTP. For details, please refer to section **Report Properties** in *SevOne NMS User Guide*.

1. In the **Server** field, enter the IP address or host name of the SFTP server where SevOne NMS is to send reports.
2. In the **Port** field, enter the port to which SevOne NMS is to send reports.
3. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the SFTP server.
4. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the SFTP server.
5. In the **Path** field, enter the path to the location on the SFTP server where you want the report to be sent.
6. Click **Test SFTP Settings** to verify that your SFTP settings work correctly.
7. Click **Save** to save the SFTP settings.

11.1.3.21 SNMP

The SNMP subtab enables you to define the SNMP settings for devices on which you enable the SNMP Plugin. You can override these settings for individual devices from the Edit Device page. For details, please refer to sections **SNMP Plugin** and **Edit Device** in *SevOne NMS User Guide*.

1. Select the **Strictly Support RFC 2233** check box to enforce strict support of RFC 2233. When the check box is selected, it means the following.
   a. for interfaces that operate at *20 Mbps or less*, 32-bit byte and packet counters must be used.
   b. for interfaces that operate *faster than 20 Mbps and slower than 650 Mbps*, 32-bit packet counters and 64-bit octet counters must be used.
   c. for interfaces that operate at *650 Mbps or faster*, 64-bit packet counters and 64-bit octet counters must be used.

   > The 64-bit counters are only used when the 32-bit counters do not provide enough capacity. When 64-bit counters are in use, the 32-bit counters must still be available. They will report the low 32-bits of the associated 64-bit count.
   > Certain combinations of Strictly Support RFC 2233 and Counter Preference can result in data loss.

   > If **Strictly Support RFC 2233** check box is not selected, it means that strict RFC 2233 Support is not used.

2. Select the **SNMP Version Lock** check box to use the version of SNMP you select. This prevents the SNMP plugin from trying to determine the proper version if the version you select fails.
3. Select the **Discover Max PDUs for Devices** check box to attempt to discover the maximum data packet size allowed by devices.

   > SNMP Protocol Data Unit, or SNMP PDU, data types are complex and specific to SNMP. The PDU field contains the body of an SNMP message. SevOne NMS uses two PDU types, **GetRequest** and **SetRequest**, which hold the necessary data to get and set parameters.

4. Click the **Counter Preference** drop-down. This setting controls how the SNMP plugin determines what counter type (32 bit or 64 bit) to choose. If you select the **Strictly Support RFC 2233** check box, this setting does not apply to in and out utilization for interfaces.

   > Certain combinations of **Strictly Support RFC 2233** and **Prefer 64-bit Counters** can result in data loss.

- **Allow Both** - use both 64-bit and 32-bit counters for an *object*.
- **Prefer 64-bit** - if interfaces are under 20Mbps, 64-bit counters are not used when 32-bit counters are available. If the interfaces are over 20 Mbps, 32-bit counters are not used when 64-bit counters are available.
- **Prefer 32-bit** - use 32-bit counters.

5. The **Synchronization Objects** section lets you specify whether to poll *objects* that are administratively or operationally down. You can override these settings on a per-*object* basis using the Object Manager. Please perform the following actions.

   > The OIDs that specify the administrative and operational status of an *object* are part of the *object type* definition.

   a. Select the **Administrative State** check box to hide and not poll objects that are administratively down. Leave clear to poll administratively down objects normally. The Object Manager enables you to override this setting on a per object basis. For details, please refer to section **Object Manager** in *SevOne NMS User Guide*.
   b. Select the **Operational State** check box to hide and not poll objects that are operationally down. Leave clear to poll operationally down objects normally. The Object Manager enables you to override this setting on a per object basis.
6. The **Default Community Strings** section displays the SNMP community strings to use during discovery. The field on the left (**Read Community Strings**) displays the list of read-community string in the sequence of precedence and the field on the right (**Write Community Strings**) displays the write strings in the sequence of precedence. When SevOne NMS discovers a device and attempts to poll SNMP data, the first string in the list is tested. If that string fails, the subsequent strings are tested, in sequence, until a string is successful. The successful community string appears on the Edit Device page for the device. For details, please refer to section **Edit Device** in *SevOne NMS User Guide*.
   a. In the **Read Community Strings** field and the **Write Community Strings** field, click **Add** to add a new row in the list.
   b. Enter the community string and click **Update**.
   c. Repeat the previous steps to add additional strings.
   d. Click the up / down arrows under **Actions** to move the string up or down in the list. The discovery process goes through the list sequentially.
7. Click **Save** to save the SNMP settings.

11.1.3.22 Storage

The Storage subtab enables you to define the size of items in the system.

> Changes to these settings can cause data loss. Please consult with your SevOne Support Engineer before you modify these settings.

1. In the **Data Retention** field, enter the number of days' worth of data to store. The default/recommended value is 365 days. Increasing this value means that the physical storage requirements will be much greater. The minimum value is 1 day and the maximum value of 730 days.

   > Data retention greater than 365 days is not supported. SevOne recommends you contact SevOne Support before you click Yes to proceed.

2. In the **Maximum Disk Utilization** field, enter the percentage of disk space to allocate for the storage of poll data (between 80 and 100 percent). The default is 95 percent, which is recommended. Leave some disk space for logs and flow data. The FlowFalcon subtab (described above) enables you to define FlowFalcon raw data retention.
3. Click **Save** to save the Storage settings.

11.1.3.23 Syslog

The Syslog subtab enables you to define where SevOne NMS is to send Syslog data. You can override the cluster level Syslog destination at the peer level from the **Peer Settings** tab described later in this topic.

1. Click **Add Syslog Destination** or click  to add or edit a Syslog destination.
2. In the **Destination Name** field, enter the name of the host/destination device to which to send the Syslog data.
3. In the **IP Address** field, enter the IP address of the host/destination device.
4. Click the **Protocol** drop-down and select **TCP** or **UDP** or **TLS** for the port type to which to send Syslog data.
5. In the **Port** field, enter the port number to which to send Syslog data.
6. Click **Update** to save the destination.
7. Repeat to add additional destinations to the list.
8. Click **Save** to save the Syslog settings.

Configure Syslog Destinations via Command Line Interface

Syslog Destinations can be created, modified, or deleted using the Command Line Interface (CLI) as well.

```
$ SevOne-act syslog-destination [create, delete, update]
```

Create Syslog Destination

To create a syslog destination, provide the following options.

Flags	Description
--uid	(Required) The id of the user executing the action.
--peer-id	(Optional) The ID of the peer for which the destination is to be created. Default: 0
--name	(Required) Unique name for the syslog destination.
--host	(Required) The host of the syslog destination.
--protocol	(Required) The protocol for the syslog communication.
--port	(Required) The port of the remote syslog.

Update (modify) Syslog Destination

To update (modify) a syslog destination, provide the following options.

Flags	Description
--uid	(Required) The id of the user executing the action.
--peer-id	(Optional) The ID of the peer for which the destination is to be updated. Default: 0
--id	(Optional) The ID of syslog destination to be updated. Default: 0
--name	(Required) Unique name for the syslog destination.
--host	(Required) The host of the syslog destination.
--protocol	(Required) The protocol for the syslog communication.
--port	(Required) The port of the remote syslog.

Delete Syslog Destination

To delete a syslog destination, provide the following options.

Flags	Description
--uid	(Required) The id of the user executing the action.

Flags	Description
--peer-id	(Optional) The ID of the peer for which the destination is to be deleted. Default: 0
--id	(Optional) The ID of syslog destination to be deleted. Default: 0

Configured Destinations

The configured syslog destination(s) are stored in MySQL table, **net.syslog_destinations**.

Example

```
+----+--------+-----------+----------+------+
| id | name   | host      | protocol | port |
+----+--------+-----------+----------+------+
| 1  | banana | 127.0.0.1 | UDP      | 100  |
+----+--------+-----------+----------+------+
```

The configured destination(s) are stored in **/etc/syslog-ng/conf.d/30-sevone-syslog-destinations.conf**, which is part of the **syslog-ng** configuration.

For the configured **syslog destination** example above, **30-sevone-syslog-destinations.conf** file contains the following .

Example: '30-sevone-syslog-destinations.conf' file

```
# 30-sevone-syslog-destinations
# This file is auto-generated by "SevOne-act syslog-destination generate-config --uid 1"
#
# DO NOT EDIT THIS FILE MANUALLY
# If you need to edit its contents use the Syslog Settings in the Cluster Manager.
destination remote-destinations-all {
    network(
        "127.0.0.1"
        transport("UDP")
        port(100)
        flags(syslog-protocol)
    );
};

log { source(s_sys); destination(remote-destinations-all); };
# END 30-sevone-syslog-destinations
```

The **root (default)** configuration can be found in **/etc/syslog-ng/syslog-ng.conf** where the following section defines the **source** of the syslog. It specifies that the appliance can take syslog from localhost with port **514**.

```
source s_sys {
    system();
    internal();
```

```
    udp(ip(127.0.0.1) port(514));
};
```

If you want the appliance to get syslog from a remote appliance or would like the appliance to receive syslog but from a different port, you may change the **protocol**, **host**, and **port number** for the source.

Please do not edit the syslog configuration file directly. SevOne recommends one of the following options.

- **Graphical User Interface** - **Administration** > **Cluster Manager** > **Cluster Settings** tab > **Syslog** subtab.
- **Command Line Interface**

```
$ SevOne-act syslog-destination
```

For additional details, please refer to **syslog-ng** documentation such as, https://www.syslog-ng.com/technical-documents/doc/syslog-ng-open-source-edition/3.22/administration-guide/12.

11.1.3.24 Topology

The Topology subtab enables you to manage which topology sources are discovered for each device type.

-  - Topology source discovered for the selected device type.
-  - Topology source not discovered for the selected device type.

You can discover topology sources independently at each level of the device type hierarchy.

1. Select a device type in the Device Types hierarchy in the field on the left.
2. Slide the toggle to enable or disable discovery of each topology source for the device type you select.
3. Repeat for each device type in the hierarchy.
4. Click **Save** to save the Topology settings.

11.1.3.25 Tracing

This feature is for **Internal Use Only** for the Support Team to use for troubleshooting.

11.1.3.26 Trap Collector

The Trap Collector subtab enables you to define the trap collector settings for devices.

Changes to these settings can cause data loss. Please consult with your SevOne Support Engineer before you modify these settings.

1. In the **Threads** field, enter the number of trap-handling threads to use. Each thread handles one trap at a time. The minimum value is 1 thread and the maximum value is 99 threads. The default value is 10 threads.
2. In the **Update Interval** field, enter the number of seconds for how often the trap collector updates the event information and caches data. The default value is 300 seconds. The minimum value is 1 second and the maximum value is 300 seconds.
3. Click **Save** to save the Trap Collector settings.

11.1.3.27 WMI Proxies

The WMI Proxies subtab enables you to define the WMI proxy servers for the WMI plugin use to poll WMI data and provides links to the WMI Proxy service and the .NET 3.5 Framework software you need to install on the proxy server. Please refer to section Enable WMI and to section **WMI Plugin** in *SevOne NMS User Guide* for details.

1. In the WMI Proxies section, click **Add WMI Proxy** or click  to add or edit a WMI proxy server.

2. In the **Name** field, enter the name of the proxy server.
3. In the **IP Address** field, enter the proxy server IP address.
4. In the **Port** field, enter the port for the proxy server to use to communicate with SevOne NMS (default – 3000).
5. Select the **Encryption Support** check box to allow support for encrypting the password. Enable the check box to allow you to enter the password in **Encryption Password** field.
6. On the pop-up, click **Save**.
7. Repeat the previous steps to define additional WMI proxy servers.
8. Click **Save** to save the WMI Proxy settings.

Downloads

The Downloads section includes the SevOne NMS WMI Proxy service file and the .NET 3.5 Framework installation file. On the WMI proxy server, run the .NET 3.5 Framework setup.exe if needed, then run the SevOne WMI Proxy Setup.msi to install the SevOne NMS WMI Proxy service.

- The SevOne NMS WMI Proxy file installs a Windows service on the Windows device you designate to act as the proxy to perform WMI queries. Click the **WMIProxy Download Installation Package** link and save the file to the proxy device.
- If the proxy device is not running the Microsoft .NET 3.5 framework, click the **.NET 3.5 Framework Download Installation Package** link to download the .NET installation package setup.exe file.

Follow the steps below to use the encryption capability for WMI traffic.

- Download the file from SevOne NMS > **Administration** > **Cluster Manager** > **Cluster Settings** > **WMI Proxies** > click on **Download Installation Package**.
- Send the downloaded file to the same server/computer where you have it previously running.
- Uninstall the WMI Proxy (if one exists).
- Install the new version of the WMI Proxy.
- Enable the Encryption Support field.
- Enter the Encryption Password.

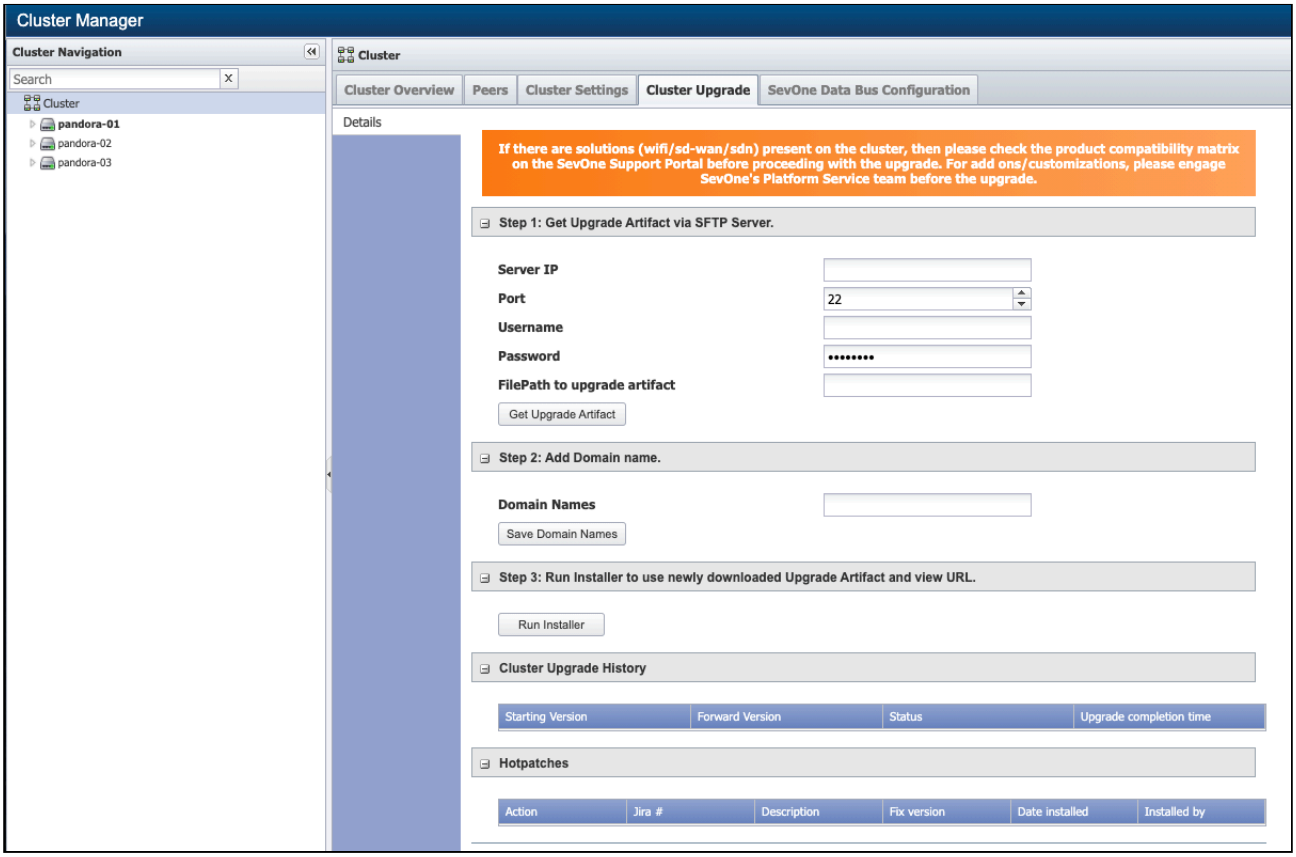
SevOne provides a Windows native proxy to ensure speed and integrity of Windows native metrics.
The functionality with encryption is:
SevOne(encryption/decryption) ↔ **Encrypted Traffic** ↔ **WMI Proxy(encryption/decryption)** ↔ **Polled Windows device** (when encryption is enabled/supported)

11.1.4 Cluster Upgrade

Self-Service Upgrades

- For Self Service Upgrades, SevOne requests the customer to raise a proactive ticket to make **SevOne Support** aware that the customer will be performing this. By doing this, SevOne Support can assist the customer with the upgrade preparation and readiness.
- Self-Service Upgrades may result in a potential IP address overlap between the customer's network and SevOne's Docker IP address range 172.17.0.0/16. If this conflicts with the customer's network, please contact **SevOne Support**.
- If there are Solutions such as SD-WAN, WiFi, and SDN present on your cluster, then please check the product **Compatibility Matrix** on **SevOne Support Customer Portal** before proceeding with the upgrade.
- For add-ons/customizations, please engage **SevOne's Platform Services** team before the upgrade.

Click  **Cluster** in the cluster hierarchy on the left and select the **Cluster Upgrade** tab on the right to upgrade the cluster using the graphical user interface. This tab will contain all the details for the SevOne NMS Graphical User Interface installer and the upgrade history.



11.1.4.1 Get Upgrade Artifact via SFTP Server

Enter the values in the following fields for the SevOne NMS being upgraded.

- Server IP - The IP Address or hostname of the SFTP server for SevOne NMS to use.
- Port - The port number on which the SFTP server is running on the remote server. The default value is port 22. SevOne NMS will send the reports to this port.
- Username - The username for copying the artifact from the remote server.
- Password - The password SevOne NMS needs to authenticate onto the SFTP server.
- FilePath to upgrade artifact - The path to the artifact on the remote SFTP server from where you wish to download the tar file. The user must have read permissions to the artifact.
- Click on Get Upgrade Artifact button to get the artifact to be used by SevOne NMS for the upgrade. The artifact is put in /opt directory of the Cluster Leader.

> If you have already configured the SFTP server on Cluster Manager > Cluster Settings tab > SFTP subtab, the same will be fetched except for the path. You may use the same server or configure a different one here.
> Depending on the size of the artifact, this step may take some time.

11.1.4.2 Add Domain Name

Enter the value in the following field for the SevOne NMS being upgraded.

- Domain Names - Enter comma separated domain names without https://. For example, test.sevone.com,test2.sevone.com.
- Click Save Domain Names button to save the domain names.

11.1.4.3 Run Installer

> IMPORTANT

Run Installer may take several minutes to respond. Please do not cancel or retry. While loading, it checks for the .tar files available for the update and also, sets the Graphical User Interface installer service.

After the upgrade artifact is downloaded, you can upgrade the installer with the latest version available in the artifact. Click on the Run Installer button and the following will be processed in the background.

- The latest installer from the artifact is extracted.
- The installer is upgraded to the latest version.
- A URL for the installer is generated.

You may proceed to the generated URL to initiate the upgrade via the Graphical User Interface.

By default, the installer runs on port 9443. However, you may change the port and reconfigure the installer to run on any port. To change the port on which the Graphical User Interface installer runs, go to Cluster Manager > Cluster Settings tab > Ports subtab. You may change the SevOne-gui-installer Port to any value desired. If cluster-wide firewall setting is enabled, this will automatically add the new port to the allowed ports list.

Login and upgrade are only allowed if you have administrative permissions.

11.1.4.4 Cluster Upgrade History

This section shows the cluster upgrade history for all the previous upgrades done using the Graphical User Interface installer. The following details are available.

- Starting Version - The SevOne NMS version of the cluster prior to the upgrade.
- Forward Version - The SevOne NMS version of the cluster that it is upgraded to.
- Status - The status of the upgrade. i.e., it indicates whether the upgrade is in progress, successful, or has failed.
- Upgrade completion time - This field shows the time it took to complete the upgrade.

11.1.4.5 Hotpatches

When you click Run Installer button, it will return the hotpatch available on the system, if any.

Example

SevOne NMS 6.0.2 is available to upgrade. To access the SevOne GUI Installer, proceed to https://10.129.14.168:9443 with admin credentials.

Click the URL and perform the Self-Service Upgrade. Please refer to *SevOne NMS Upgrade Process Guide* > section Self-Service Upgrade for details.

If there is no hotpatch available on the system, you will see a message as shown below in the example.

Example

No upgrade available. To access the SevOne GUI installer, proceed to https://10.129.14.168:9443 with admin credentials.

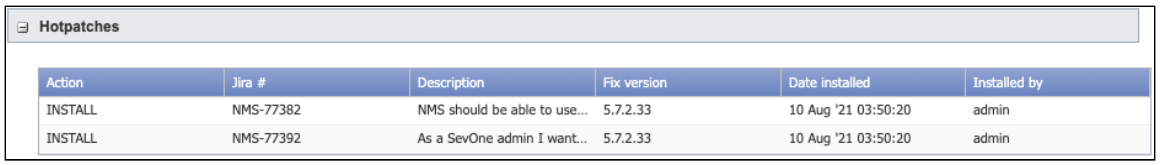
Hotpatches are cumulative. For example, lets say there are two hotpatches, 6.0.1 and 6.0.2. If 6.0.1 contains a fix for A, 6.0.2 must contain the fix for A and B.

The following details are available.

- Action - informs the action performed with the hotpatch. It can be an action to install or revert.
- Jira # - provides the Jira ticket number to reference to for details.
- Description - provides the description.
- Fix version - the version in which the fix is made generally available.

- **Date installed** - provides the date when the installation was performed.
- **Installed by** - provides the name of the person who performed the action. For example, **admin**.

Example

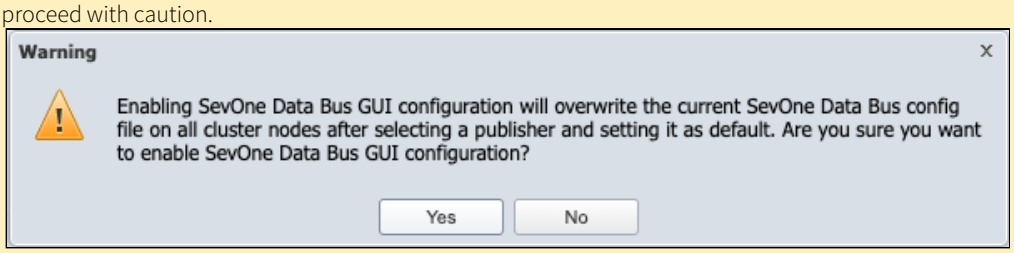


11.1.5 SevOne Data Bus Configuration

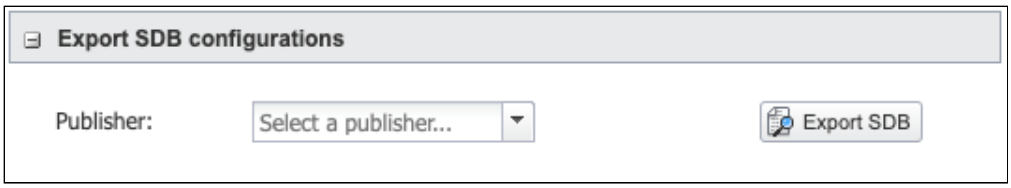
SevOne Data Bus configuration using the Graphical User Interface:
- does not support SevOne Data Bus' historical replay
- backs-up existing configuration file on upgrade; any other configurations are not backed-up
- writes/overwrites **/etc/sevone/data-bus/application.conf** file and **/etc/sevone/data-bus/databusmsg.json** file on each node in the cluster

1. Select the **Enable SevOne Data Bus GUI configuration** check box to allow user to configure SevOne Data Bus through the interface and generate a configuration file.

   By selecting the check box, the following warning message will appear. Please read the warning message and proceed with caution.

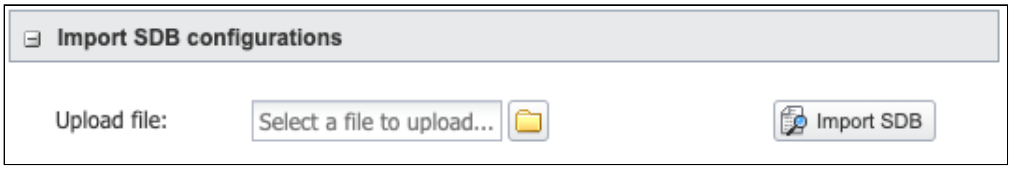
   

2. If you click Yes in the Warning message above, the following configuration capabilities become available.
   a. **Export SDB configurations**

      i. Click the **Publisher** drop-down and select a publisher from the list.
      ii. Click **Export SDB** to allow the *admin* to export existing SevOne Data Bus configurations as a downloadable file. For example, **SDB Configurations.spk** file.

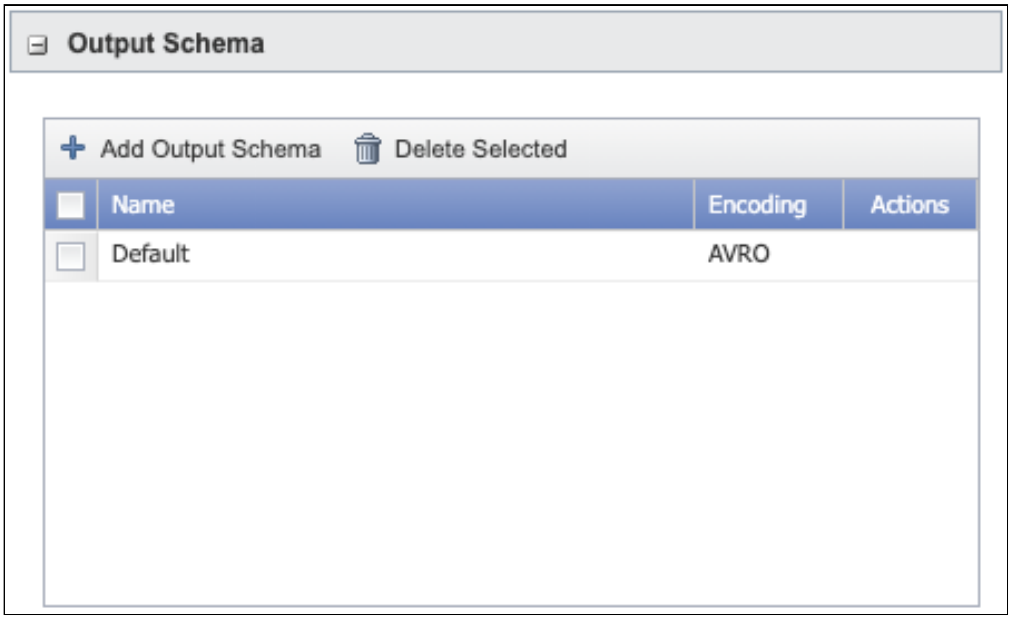
   b. **Import SDB configurations**



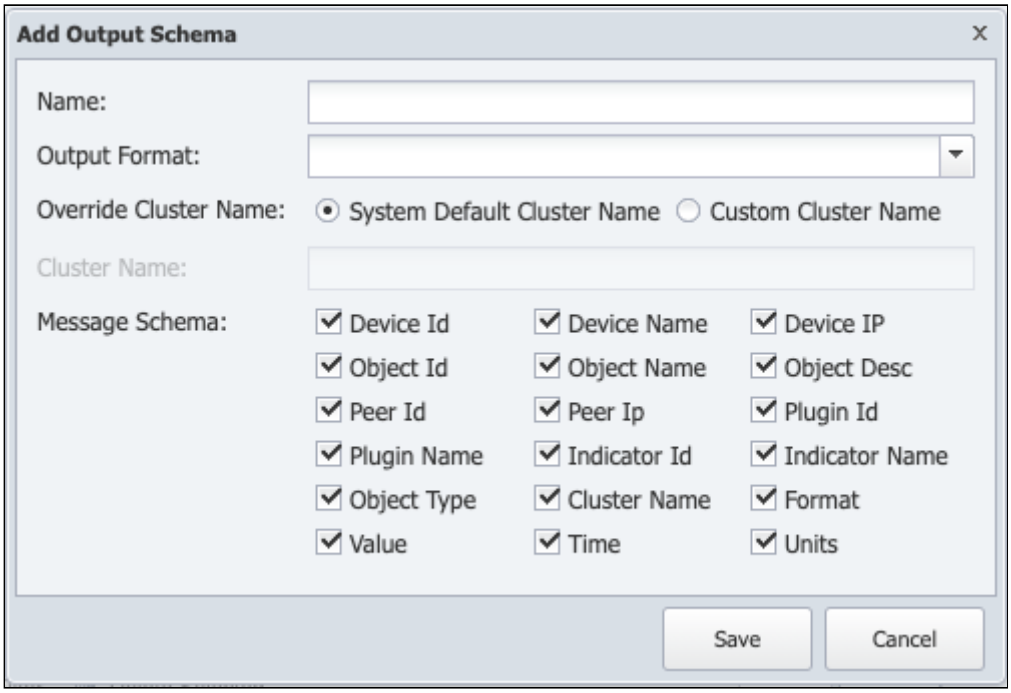
i. **Upload file** - click to import the SevOne Data Bus file (**.spk** file) from the directory where it can be uploaded from.
ii. Click **Import SDB** to allow the *admin* to import a file containing SevOne Data Bus configurations to a cluster.z

> If the same file is uploaded again or if a file with the same name is uploaded, the imported file is created with a unique name. For example, **SDB Configurations.spk** becomes **SDB Configurations (1).spk**.

c. **Output Schema**



By default, an output schema is available with output format, **AVRO**, and all indicator fields selected. For additional schemas, click **Add Output Schema**.

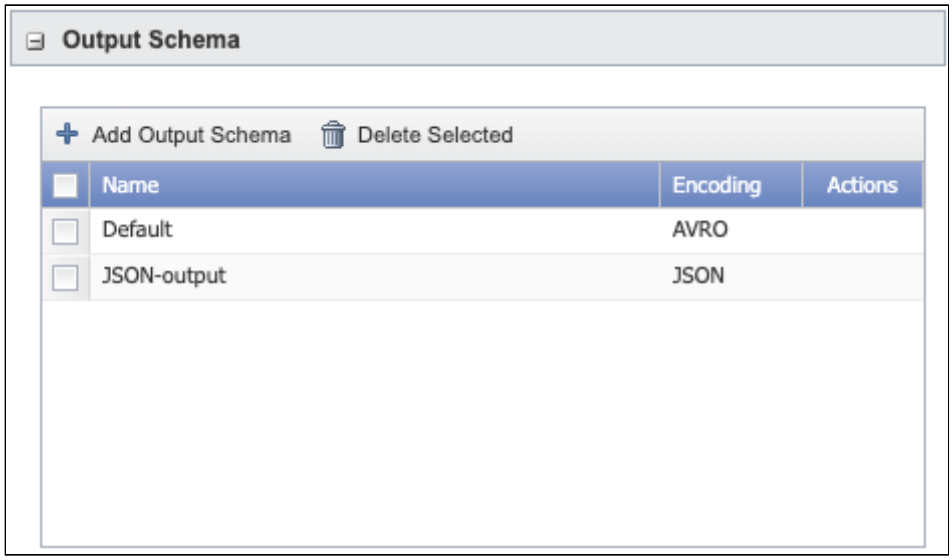


i. In the **Name** field, enter the name for the output schema.
ii. Click the **Output Format** drop-down and select one of the following options.
   - **AVRO** - When using avro, you can configure the JSON schema to customize the fields that SevOne Data Bus exports.
   - **JSON** - When using JSON, all of the message schema fields are exported.

   > ⚠ Fields **Cluster Name** and **Message Schema** are only available when **Output Format** selected is **AVRO**. Schema file is not used for output format, **JSON**.

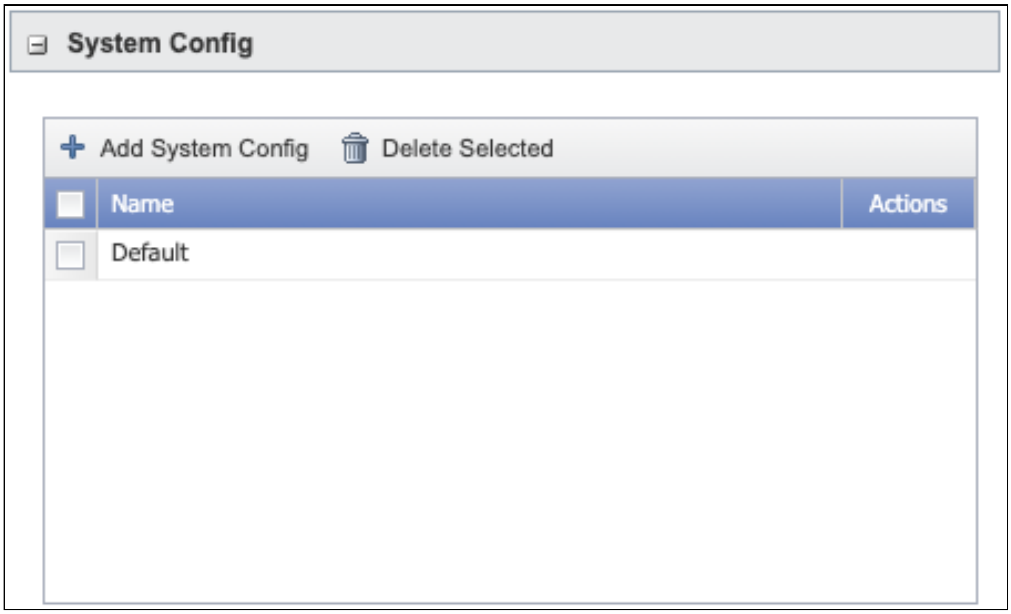
iii. Select an option for **Override Cluster Name**.
   - Select **System Default Cluster Name** option to use the system cluster name.
   - Select **Custom Cluster Name** option to enter a custom cluster name in field **Cluster Name**.
iv. By default, all indicators are selected in field **Message Schema**. You may choose to unselect one or more indicators.
v. Click **Save** to save the output schema.

Example

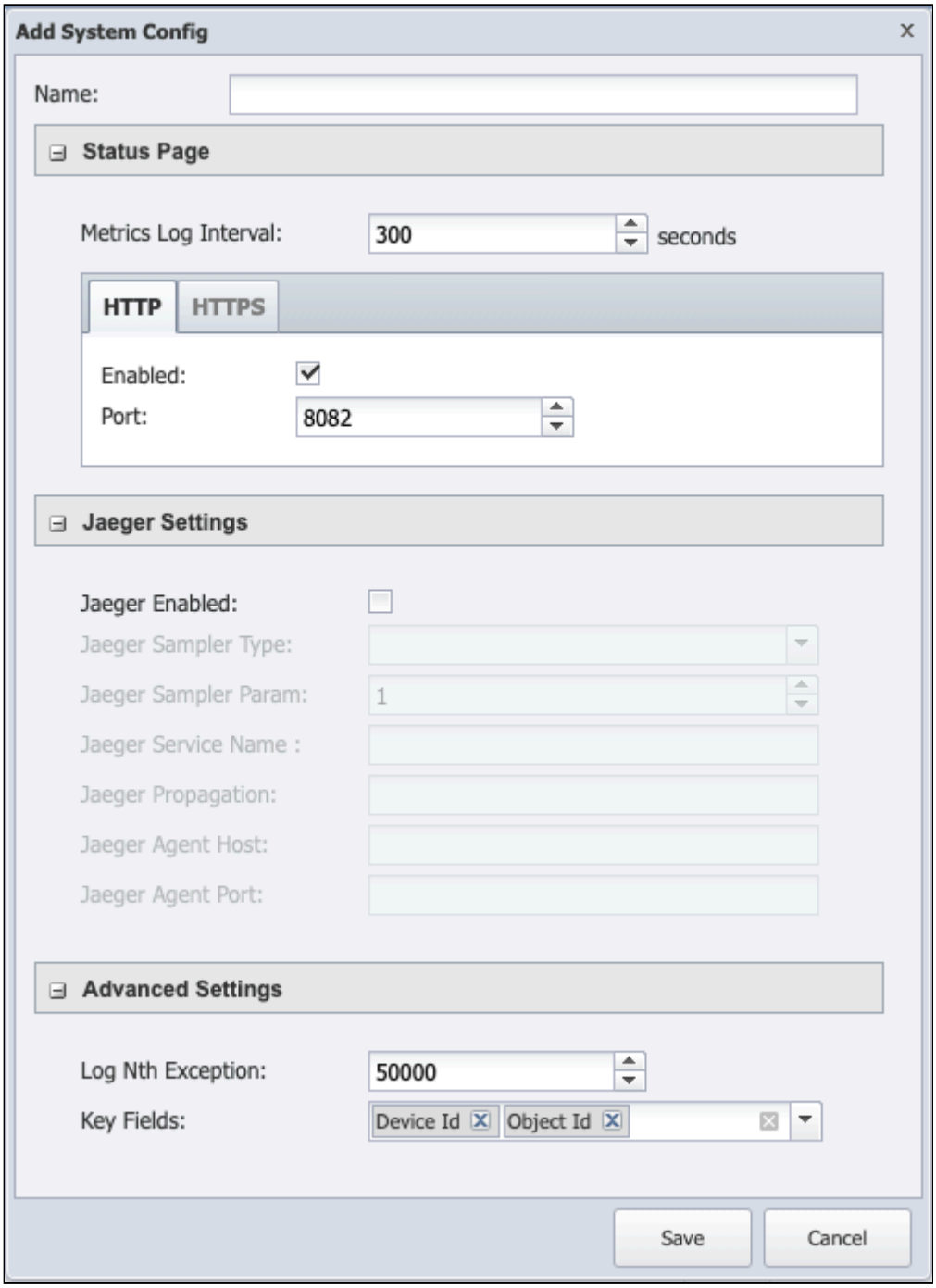


To delete an output schema, click **Delete Selected**.

d. **System Config**



A system configuration is created by default. For additional system configurations, click **Add System Config**.



i. In the **Name** field, enter the name for the system configuration.
ii. **Status Page**
   - In the **Metrics Log Interval** field, enter the number of seconds for how often the metrics must be updated. The minimum value is 60 seconds and the maximum value is 360000 seconds. The default value is 300 seconds.
   - Under tab **HTTP** (default),
     - Select the **Enabled** check box to enable HTTP status page.
     - In the **Port** field, enter the port number SevOne Data Bus status page runs on. The default port is **8082**.
   - Under tab **HTTPS**,
     - Select the **Enabled** check box to enable HTTPS status page.

- In the **Port** field, enter the secure port that the SevOne Data Bus status page runs on. The default port is **8443**.
- In the **Private Key Password** field, enter the private key password.
- In the **Keystore Password** field, enter the keystore password.
- In the **Keystore Path** field, enter the path to the keystore.

iii. **Jaeger Settings** - SevOne Data Bus supports OpenTracing. The following Jaeger environment values are supported. For additional details, please refer to https://www.jaegertracing.io/docs/1.22/sampling/.
- Select the **Jaeger Enabled** check box to enable Jaeger tracing.
- Click the **Jaeger Sampler Type** drop-down and select the type of sampling Jaeger will perform.
  - **CONST** - Sampler always makes the same decision for all traces. It either samples all traces (Jaeger Sampler Param=1) or none of them (Jaeger Sampler Param=0).
  - **PROBABILISTIC** - Sampler makes a random sampling decision with the probability of sampling equal to the value entered in **Jaeger Sampler Param** field. For example, if Jaeger Sampler Param is set to 0.1, approximately 1 in 10 traces (10%) of traces will be sampled.
  - **RATELIMITING** - Sampler uses a leaky bucket rate limiter to ensure that traces are sampled with a certain constant rate. For example, when Jaeger Sampler Param is set to 2.0, it will sample requests with the rate of 2 traces per second.
  - **REMOTE** - Sampler consults Jaeger agent for the appropriate sampling strategy to use in the current service. This allows controlling the sampling strategies in the services from a central configuration in Jaeger backend.
- In the **Jaeger Sampler Param** field, enter a number to control the sampler. The minimum value is 0 and the maximum value is 10. The default is **1**. Please refer to **Jaeger Sampler Type** field above for details.
- In the **Jaeger Service Name** field, enter the user-supplied service name that will be associated with emitted spans.
- In the **Jaeger Propagation** field, enter the propagation format used by the trace. The supported values are **jaeger**, **b3**, and **w3c**.
- In the **Jaeger Agent Host** field, enter the IP address or hostname where the spans will be reported. The default is **localhost**.
- In the **Jaeger Agent Port** field, enter the UDP port on Jaeger Agent Host where the spans will be reported. For example, **6831**. For details on port numbers required for tracing, please refer to *SevOne NMS Port Number Requirements Guide*.

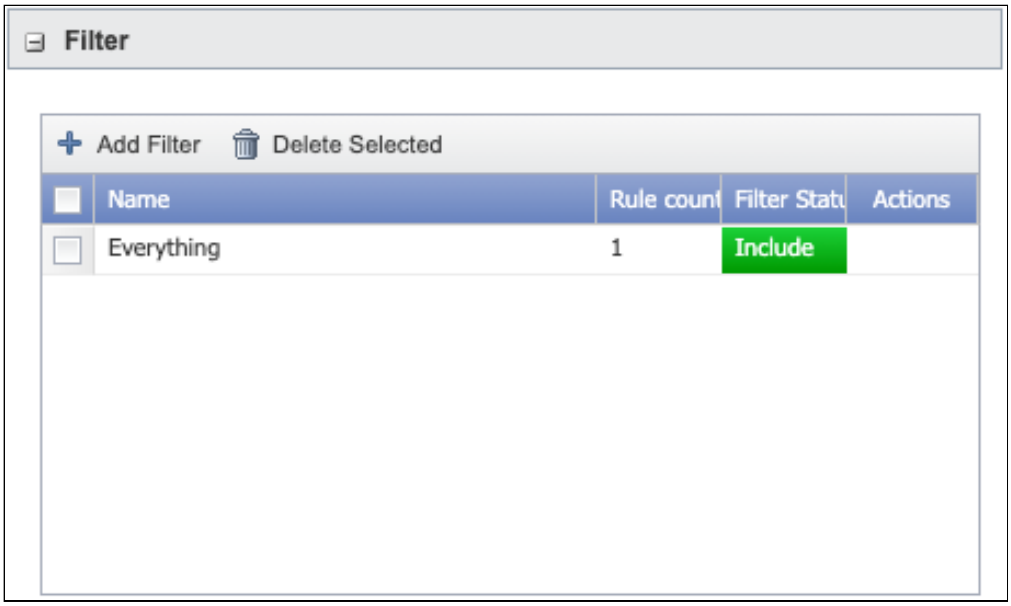
iv. **Advanced Settings**
- In the **Log Nth Exception** field, enter a number to log only every Nth consecutive exception when *publishing* errors occur. The default is **50000**.
- Click the **Key Fields** drop-down to select one or more indicators to use with Kafka hashing. The default is **Device Id, Object Id**.

v. Click **Save** to save the system configuration.

Example

To delete a system configuration, click **Delete Selected**.

e. **Filter**



By default, a filter named **Everything** is created. For additional filters, click **Add Filter**.

i. In the **Name** field, enter the name for the filter.

> **About filters and rules**
>
> - There are two classes of filter in SDB - **Include** filters (allowlist) and **Exclude** filters (blocklist).
> - A filter consists of a name and a list of rules.
> - A rule consists of a group of 5 attributes.
>   - Device Group ID (devGrpID)

    - Object Group ID (objGrpID)
    - Device ID (devID)
    - Object ID (objID)
    - Plugin ID (pluginID)
- The attribute value is either the ID of the particular attribute or -1 indicating that all IDs are matched. By default, the attribute value is -1. If an attribute is not specified in a rule, its default value is assumed.
- Within the **rule**, the attributes are combined in a **logical AND** operation.
  For example, the rule {devGrpID = 4, objID = 7} is the same as {devGrpID = 4, objGrpID = -1, devID = -1, objID = 7, pluginID = -1} and means *indicators with device group 4 and object ID 7 will be matched.*
- Within the **rule list**, the rules are combined in a **logical OR** operation.
  For example, the rule list [{devID=5},{devID=6, pluginID=3}] is the same as [{devGrpID = -1, objGrpID = -1, devID = 5, objID = -1, pluginID = -1},{devGrpID = -1, objGrpID = -1, devID = 6, objID = -1, pluginID = 3}] and means *indicators with device ID 5 or indicators with device ID 6 and pluginID 3 will be matched.*
- Exclude filters are applied first to remove indicators that match the filter, then the include filters are applied to select matches from the remaining indicators.

ii. Click the **Filter Status** drop-down and select one of the following options.
    - **Include** - For allowlist filter rules.
    - **Exclude** - For blocklist filter rules.
iii. Click the **Device Group** drop-down and select the device group.
iv. Click the **Object Group** drop-down and select the device group.
v. Click the **Plugin** drop-down and select the device group.
vi. Click the **Device** drop-down and select the device. Based on the device selected, you can choose an object from the **Object** drop-down.
vii. Click **Add Rule** to add a new rule.
viii. Click **Save** to save the filter.

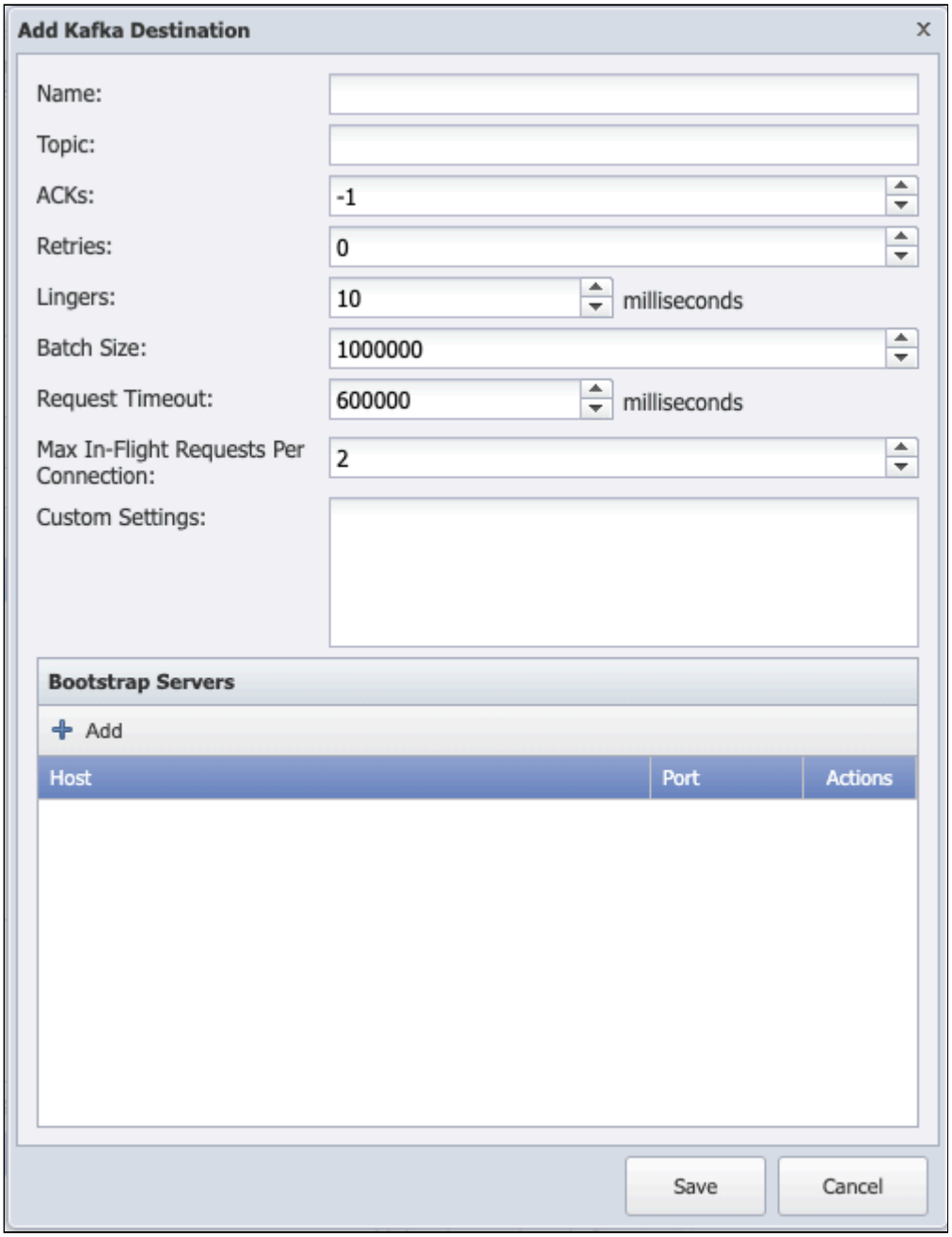
Example

To delete a filter, click **Delete Selected**.

f. **Destination**

To add Kafka destination, click Add Destination from Kafka tab to set the Kafka producer configuration settings.

Add Destination for Kafka



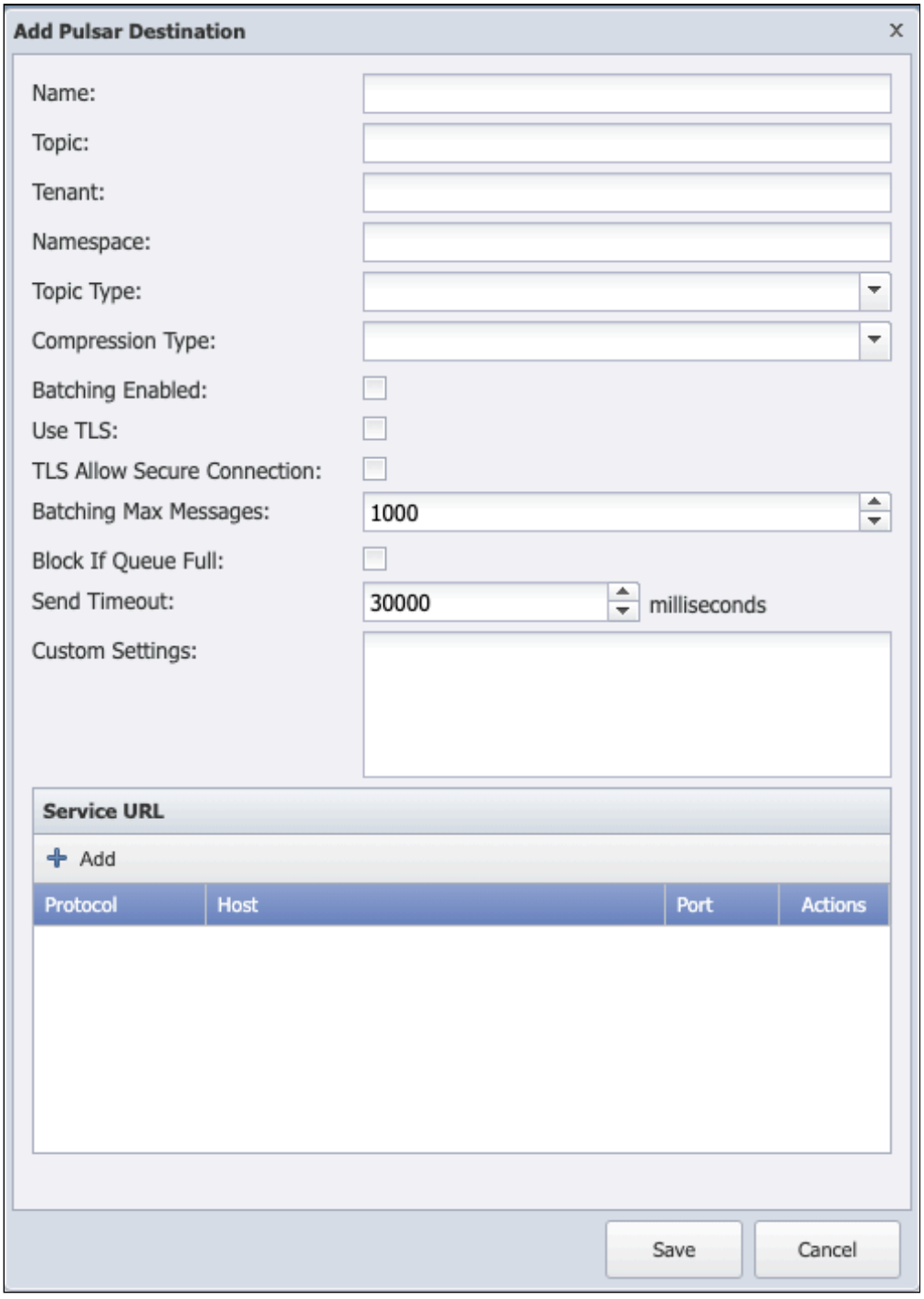
i. In the **Name** field, enter the name for the Kafka destination.
ii. In the **Topic** field, enter the name of the Kafka topic that SevOne Data Bus writes to. For example, sdb.
iii. In the **ACKs** field, select the number of acknowledgements that the *leader* must receive before a request is considered complete. The default is **-1** and is considered to be the most robust, albeit slowest, option. The available values are -1, 0, and 1. For additional details, please refer to https://kafka.apache.org/documentation/#producerconfigs_acks.
iv. In the **Retries** field, select the number of times to retry sending a failed message. The default is **0**. The minimum value is 0 and the maximum value is 100.
v. In the **Lingers** field, enter the amount of time in **milliseconds** for messages to remain in the producer queue before message batches are created. The default is **0 milliseconds**. The minimum value is 0 milliseconds and the maximum value is 300 milliseconds.
vi. In the **Batch Size** field, enter the number of messages in the batch. The default is **1000000**. The minimum value is 1000 and the maximum value is 9999999.

vii. In the **Request Timeout** field, enter the amount of time in **milliseconds** that the client will wait for a request response. The default is **600000 milliseconds**. The minimum value is 1000 milliseconds and the maximum value is 1800000 milliseconds.
viii. In the **Max In-Flight Requests Per Connection** field, enter the maximum number of unacknowledged requests sent to a broker. The default is **2**. The minimum value is 1 and the maximum value is 10.
ix. In the **Custom Settings** field, enter additional settings that are passed through to Kafka. For additional details, please refer to https://kafka.apache.org/documentation/#producerconfigs.
x. To add **Bootstrap Servers**, click **Add**. Enter the hostname or IP address in the **Host** field (for example, 10.129.13.10) and enter the port number in the **Port** field. Port **TCP 9092** is the default port number. Click **Update** to add.
xi. Click **Save** to save the Kafka destination.

To delete a Kafka destination, from **Kafka** tab, click **Delete Selected**.

To add Pulsar destination, click **Add Destination** from **Pulsar** tab.

Add Destination for Pulsar



i. In the **Name** field, enter the name for the Pulsar destination.
ii. In the **Topic** field, enter the name of the Pulsar topic that SevOne Data Bus writes to. For example, sdb.
iii. In the **Tenant** field, enter the Pulsar service tenant name.
iv. In the **Namespace** field, enter the Pulsar service namespace.
v. Click the **Topic Type** drop-down and select one of the following options.

> The Topic Type indicates whether the Pulsar broker stores messages on **persistent** storage for later consumption or stores messages in **non-persistent** storage.

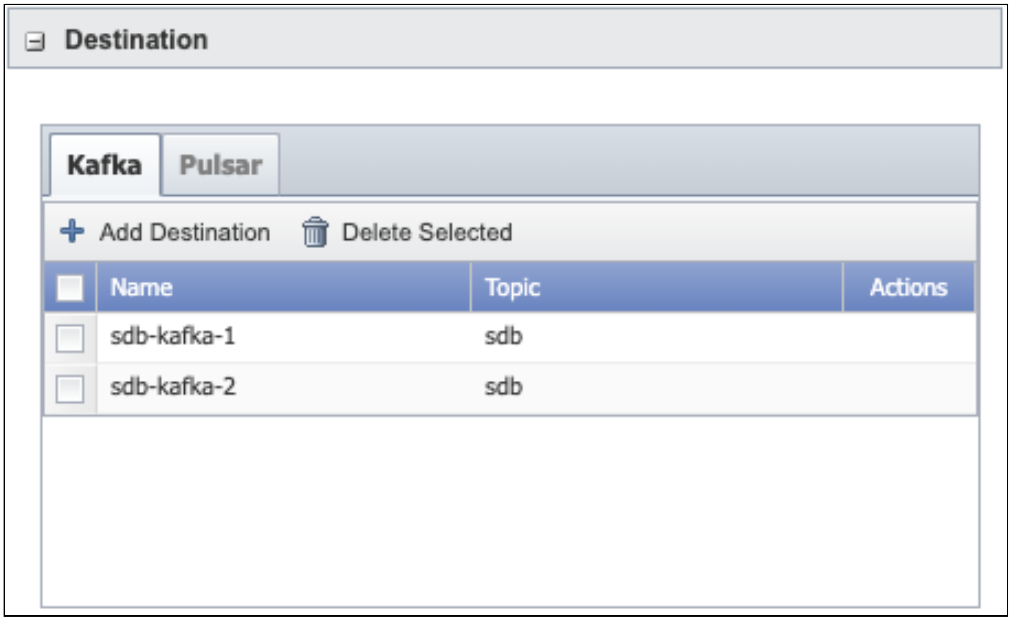
- **Persistent** - The messages are stored in the secondary storage (disk, SSD, etc.). There is some cost in terms of overhead and latency, but messages will be present if the broker is restarted.
- **Non Persistent** - The messages are stored in the primary storage (RAM). It offers higher performance for real-time messages at the cost of lost messages when the broker is restarted.

vi. Click the **Compression Type** drop-down and select one of the following options to set the compression type for the producer.
- ZLIB
- LZ4
- ZSTD
- SNAPPY

vii. Select the **Batching Enabled** check box to enable batching.
viii. Select **Use TLS** check box to use TLS.
ix. Select **TLS Allow Secure Connection** check box to allow a secure TLS connection.
x. In the **Batching Max Messages** field, enter the maximum number of messages permitted in a batch. The default is **1000**. The minimum value is 300 and the maximum value is 9999.
xi. Select **Block If Queue Full** check box for **send** operations to block when the outgoing message queue is full. For additional details, please refer to the following links.
- http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/Producer.html#send-T-
- http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/Producer.html#sendAsync-T-

xii. In the **Send Timeout** field, enter the amount of time in **milliseconds** for which Pulsar will wait to report an error if a message is not acknowledged by the server. The default is **30000 milliseconds**. The minimum value is 18000 milliseconds and the maximum value is 1000000 milliseconds.
xiii. In the **Custom Settings** field, enter additional settings that are passed through to Pulsar.
xiv. To add **Service URL**, click **Add**. By default, **Protocol** is **pulsar+ssl**. Enter the hostname or IP address in the **Host** field (for example, 10.129.13.10) and enter the port number in the **Port** field. Port **TCP 6651** is the default port. Click **Update** to add.
xv. Click **Save** to save the Pulsar destination.

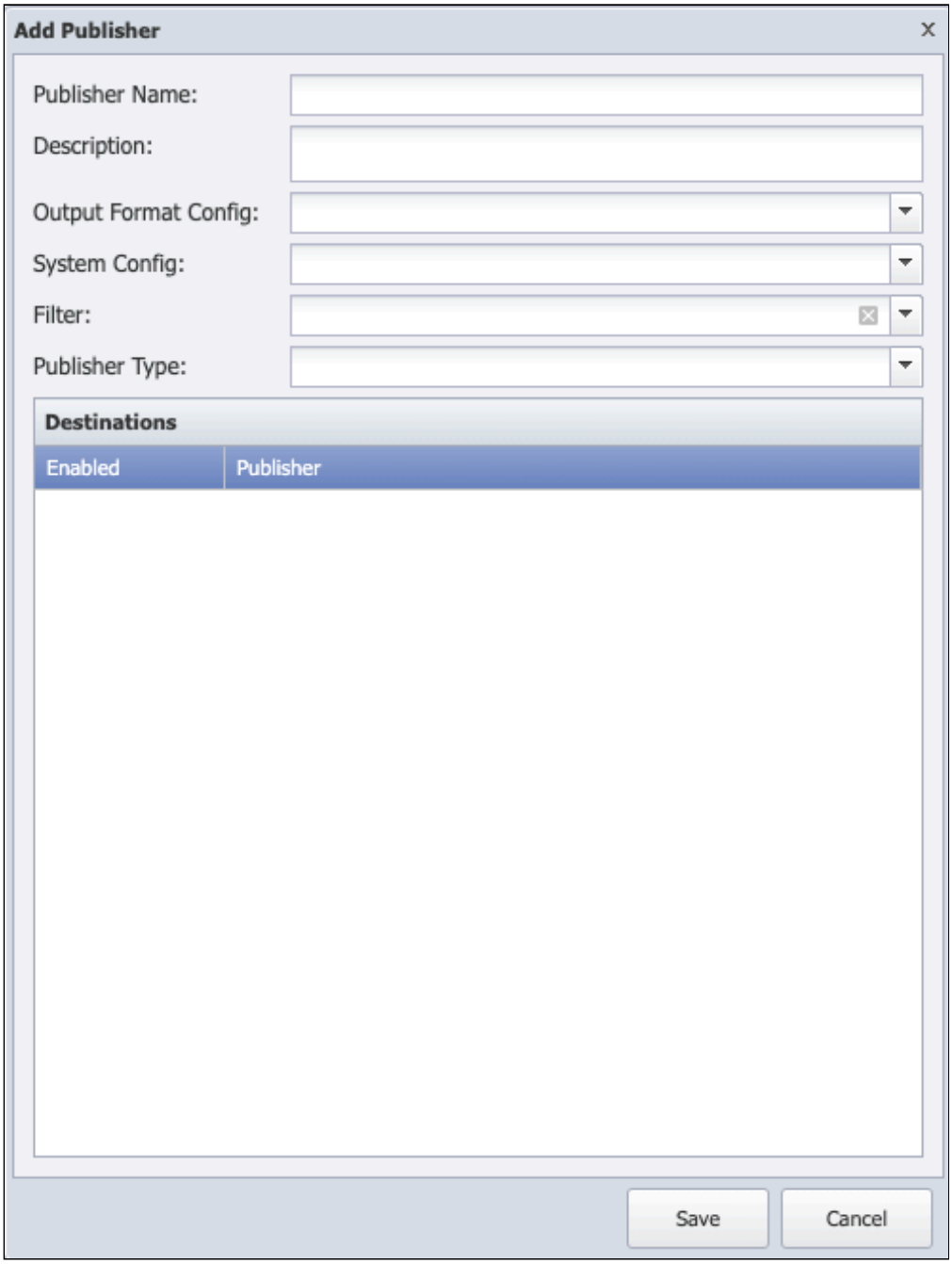
Example



To delete a Pulsar destination, from **Pulsar** tab, click **Delete Selected**.

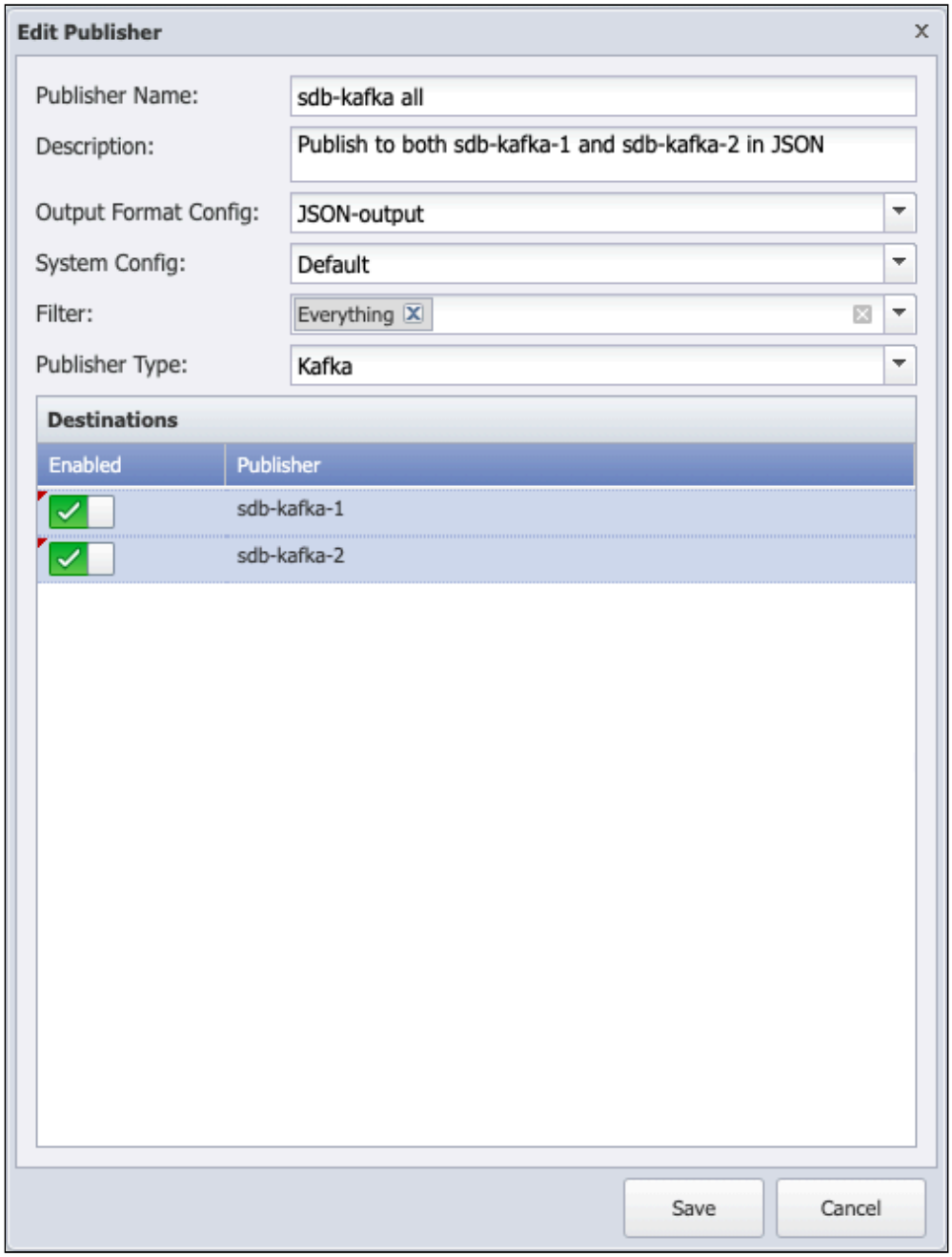
g. **Publisher**

After **Output Schema**, **System Config**, **Filter**, and **Destination** are configured, you are now ready to add a publisher. Click **Add Publisher**.



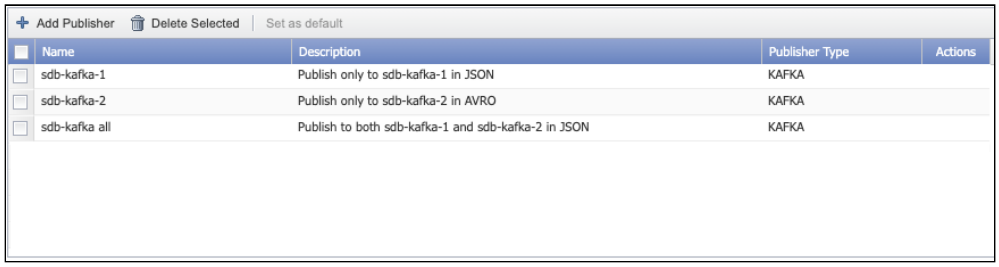
i. In the **Publisher Name** field, enter the name for the publisher.
ii. In the **Description** field, enter the description for the publisher being added.
iii. Click the **Output Format Config** drop-down and select one from a list of **Output Schemas** available.
iv. Click the **System Config** drop-down and select one from a list of **System Configurations** available.
v. Click the **Filter** drop-down and select one from a list of **Filters** available.
vi. Click the **Publisher Type** drop-down and select one of the following options.
    - **Kafka**
    - **Pulsar**
vii. Once all the fields are entered, it will provide you with the list of destinations defined. **Enable** one or more destinations from the list.

Example



viii. Click **Save** to save the publisher.

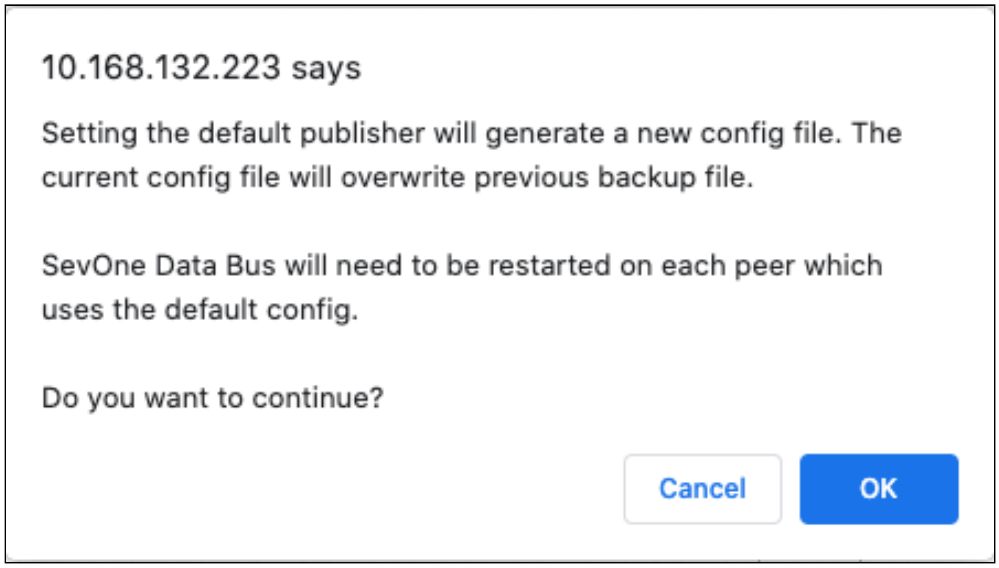
Example



To delete a publisher, click **Delete Selected**.

You are now ready to select a publisher from the list.

i. Select a publisher.
ii. **Set as default** button becomes available.
iii. Click **Set as default** button and you will get the following pop-up.



Click **OK** to continue.

SevOne Data Bus configuration has been generated. To apply this configuration, SevOne Data Bus must be restarted. To restart SevOne Data Bus, click **<peer name>** in the cluster hierarchy in the left navigation bar. Select the **Peer Settings** tab > **SevOne Data Bus** subtab. Please follow the steps in section SevOne Data Bus to restart.

11.2 Peer Level - Peer Overview and Peer Settings

<peer name> - Select a peer in the hierarchy on the left side of the Cluster Manager. The cluster leader peer name displays at the top of the peer hierarchy in bold font and other peers display in alphabetical order.

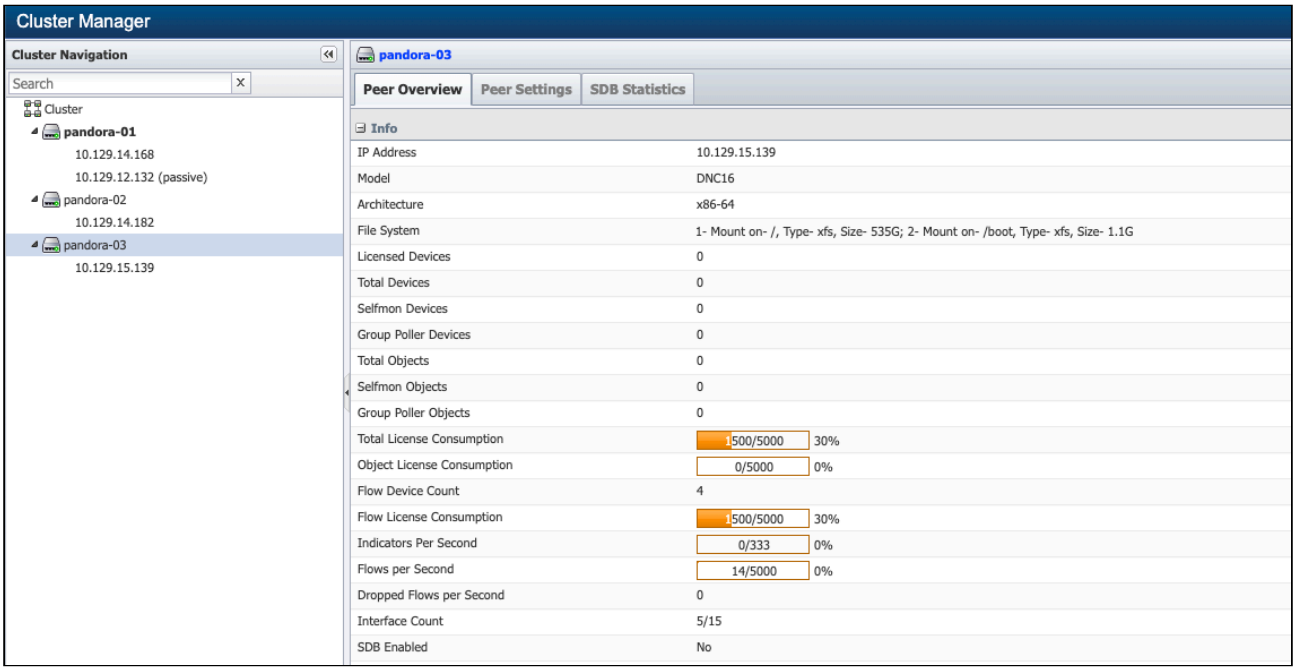
The following tabs appear on the right side to enable you to view peer level information and to define peer level settings.

- **Peer Overview** - Enables you to view peer level information.
- **Peer Settings** - Enables you to define settings that are peer specific.

Click on the peer name that displays above the Peer Overview tab to display a pop-up that enables you to rename the peer.

11.2.1 Peer Overview

Click <peer name> in the cluster hierarchy on the left and select the Peer Overview tab on the right to view peer level information.



- IP Address - Displays the IP address of the peer.
- Model - Displays the actual SevOne NMS appliance model: PAS = Performance Appliance Solution, DNC = Dedicated NetFlow Collector, vPAS = Virtual Performance Appliance Solution. For example, PAS5K, PAS60K, ..., PAS200K, DNC1000, etc.
- Architecture - Displays the architecture used for the peer. For example, x86-64.
- File System - Displays the file system you are on. The preferred file system is XFS built on CentOS. It provides the ability to detect undesirable file systems so that you can migrate / rebuild to use a more supportable file system.
- Licensed Devices - Displays the number of devices the peer discovers and polls. The Device Manager enables you to manage devices. The Licensed Devices count is equal to (Total Devices - (Selfmon Devices + Group Poller Devices)) in the peer. For details, please refer to section Device Manager in *SevOne NMS User Guide*.
- Total Devices - Displays the total number of Licensed, Selfmon, and Group Poller devices in the peer.
- Selfmon Devices - Displays the number of Selfmon devices in the peer.
- Group Poller Devices - Displays the number of Group Poller devices in the peer.
- Total Objects - Displays the total number of objects polled from the selected peer along with Selfmon and Group Poller objects. The Object Types page, Object Rules page, and Object Manager enable you to manage the number of polled objects.
- Selfmon Objects - Displays the number of Selfmon objects in the peer.
- Group Poller Objects - Displays the number of Selfmon objects in the peer.
- Total License Consumption - Displays the sum usage of objects and flow. This displays the number of flows and objects the peer is licensed to use and the percentage of the license capacity your peer uses.
- Object License Consumption - Displays the number of objects the peer uses, the number of objects the peer is licensed to use, and the percentage of the license capacity the peer uses.
- Flow Device Count - Displays the number of flow device count in the peer.
- Flow License Consumption - Displays the number of flows the peer uses, the number of flows the peer is licensed to use, and the percentage of the license capacity the peer uses.
- Indicators Per Second - Displays the total number of indicators the peer receives per second from all objects and interfaces along with backfilled data from *net.deviceipsinfo* table.
- Flows per Second - Displays the total number of flows the peer receives per second from all interfaces.
- Dropped Flows per Second - Displays the number of flows dropped per second in the peer.
- Interface Count - Displays the flow interfaces available and its capacity in a peer.
- SDB Enabled - Displays whether SevOne Data Bus is enabled or disabled for the peer.

Processed Flows + Dropped Flows = Number of Flows licensed (because these statistics are rolling averages, the total may be off slightly). The flow statistics display a weighted, rolling average of the flow data over the past hour, before duplication*, along with the

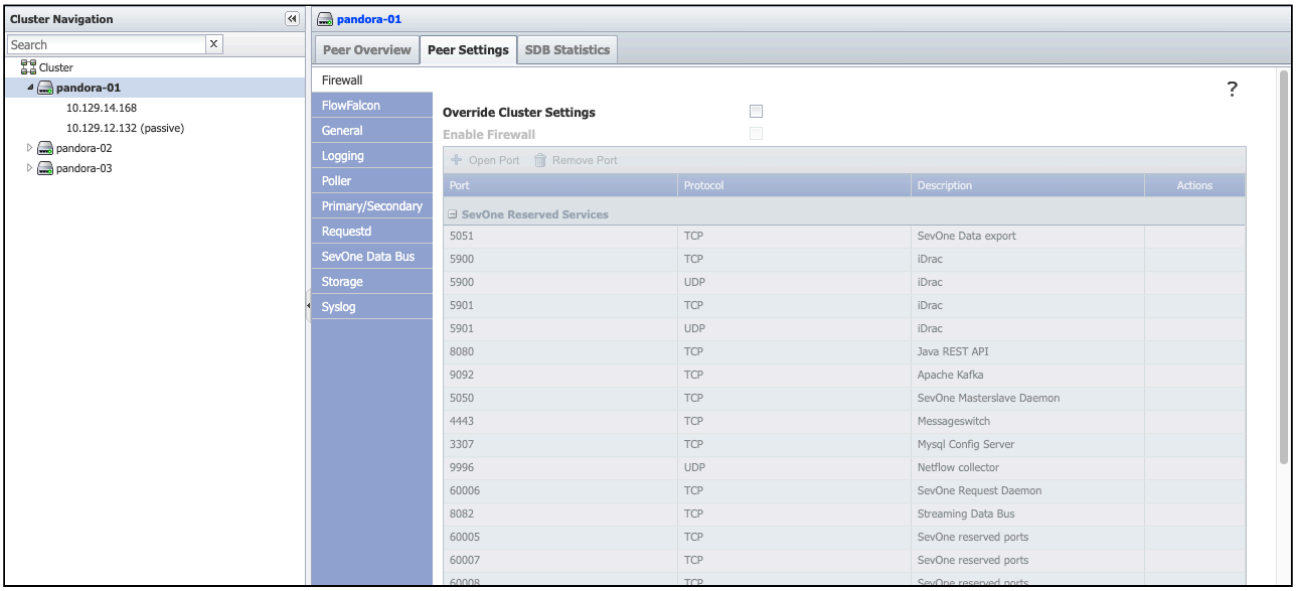
number of processed flows to assist with peer capacity management. The processed flow data does not factor in malformed flows nor the flows you deny via a rule on the Flow Rules page.

*Several pages display flow statistics. The flow statistics that each page displays are used for different purposes. Each page uses a different way to calculate flow data, mainly because v5 NetFlow only exports information about the incoming interface. SevOne NMS duplicates the flow statistics for v5 NetFlow to factor for outgoing flows on devices that use v5 NetFlow in reports but does not duplicate flow statistics for v5 NetFlow for license object consumption.

- The Cluster Manager calculates flow data without duplication for v5 NetFlow and uses a one hour rolling average.
- The Flow Interface Manager duplicates v5 NetFlow and displays the flow data for the past one minute.
- FlowFalcon reports duplicate v5 NetFlow and calculate flow data based on the report settings. For details, please refer to section **FlowFalcon Reports** in *SevOne NMS User Guide*.

11.2.2 Peer Settings

Click <peer name> in the cluster hierarchy on the left and select the **Peer Settings** tab. Subtabs appear along the left side of the Peer Settings tab to enable you to define peer specific settings.



11.2.2.1 Firewall

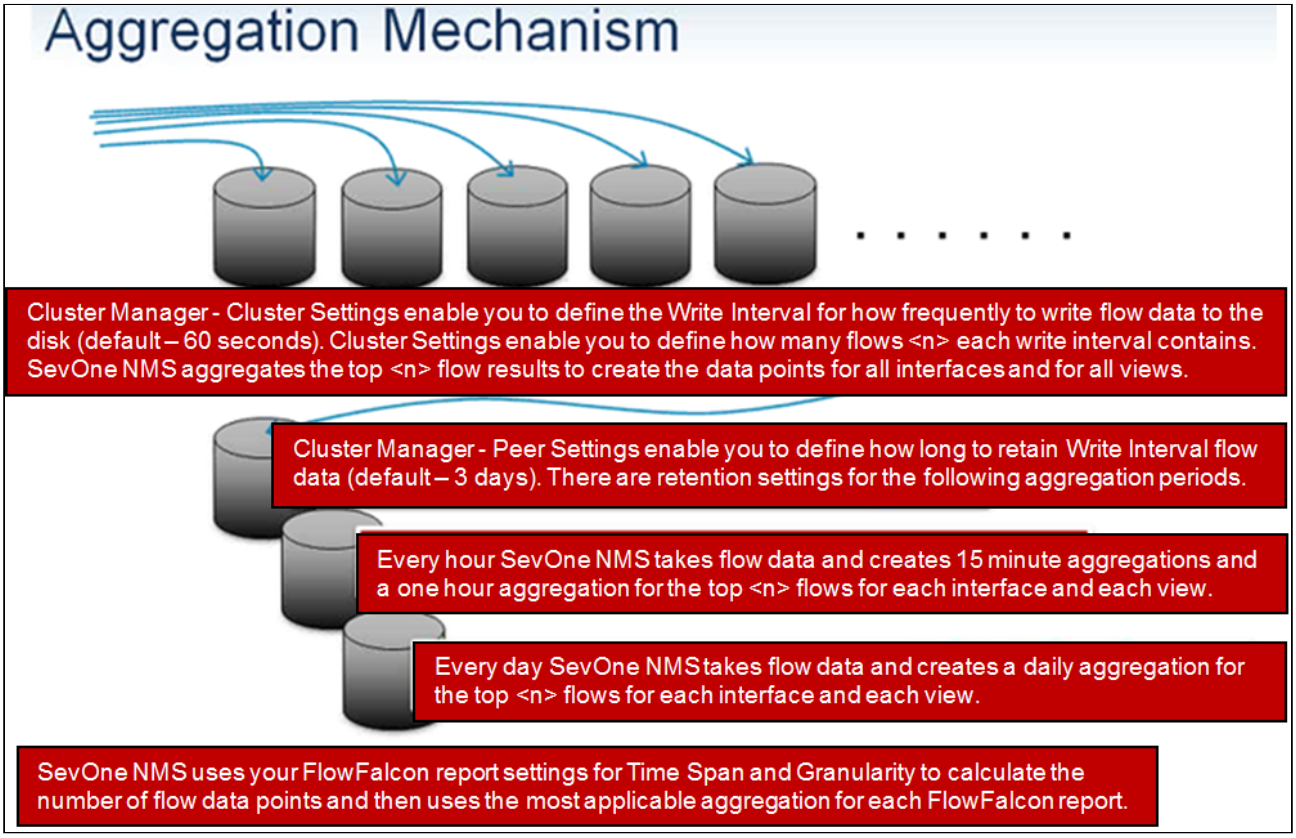
The Firewall subtab enables you to select the firewall service for the selected peer.

Select the **Override Cluster Settings** check box to override cluster-level firewall settings with firewall settings at the selected peer-level. When Override Cluster Settings field is enabled, **Enable Firewall** field is available. Click the check box to enable the firewall service for the selected peer. It is disabled by default.

Also, **Open Port** becomes available. Click on Open Port to add the firewall port for the selected peer and **Remove Port** removes user-added ports only.

11.2.2.2 FlowFalcon

The FlowFalcon subtab enables you to define the retention of aggregated flow data on the peer for use in FlowFalcon reports. You define raw flow duration on the Cluster Manager at the cluster level as described earlier in this topic. For details, please refer to section **FlowFalcon Reports** in *SevOne NMS User Guide*.



Changes to these settings can cause data loss. Please consult with your SevOne Support Engineer before you modify these settings.

1. In the **Write Interval Duration** field, enter the number of days' worth of <write interval> aggregated flow data to store for calculations. The default is 3 days. The <write interval> is defined in **Cluster Settings** tab > **FlowFalcon** subtab. Please see details above.
2. In the **Fifteen Minutes** field, enter the number of days' worth of fifteen minute aggregation data to store for calculations. Every hour, SevOne NMS takes the flow data and creates one 1 hour aggregation data points for each of the top flows for each interface and each view. The default value is 7 days. The minimum value is 0 days.
3. In the **One Hour** field, enter the number of days' worth of one hour aggregation data to store for calculations. Every hour, SevOne NMS takes the flow data and creates four 15 minute aggregation data point for each of the top flows for each interface and each view. The default value is 90 days. The minimum value is 0 days.
4. In the **One Day** field, enter the number of days' worth of one day aggregation data to store for calculations. Every hour, SevOne NMS takes the flow data and creates one 1 day aggregation data point for each of the top flows for each interface and each view. The default value is 365 days. The minimum value is 0 days.
5. Click **Save** to save the FlowFalcon peer settings.

11.2.2.3 General

The General subtab enables you to add a tunneling proxy server for each peer to use for HTTP poll requests and proxy information for VMware requests. This subtab also enables you to schedule when the peer is to perform the automatic discovery function.

1. In the **HTTP Proxy** section, in the **HTTP Proxy Server** field, enter the full URL of the HTTP server you want the peer to poll for data from devices on which you enable the HTTP plugin. This field is applicable when your implementation includes a HTTP proxy server and the URL must have a valid format with a port number. For details, please refer to section **HTTP Plugin** in *SevOne NMS User Guide*.

   Example

   http://www.yourproxyserver.com:portnumber/

2. In the **VMware Proxy** section, the following fields enable you to define how peers communicate with each other to collect VMware data from the VMware plugin. For details, please refer to section **VMware Plugin** in *SevOne NMS User Guide*.
    a. In the **Port** field, enter the port on the proxy for the peer to use to collect the VMware data from other peers.
    b. In the **Username** field, enter the user name the peer needs to authenticate onto the proxy.
    c. In the **Password** field, enter the password the peer needs to authenticate onto the proxy. The password must be <= 8 characters long.
3. In the **Automatic Discovery** section: the following fields enable you to schedule when to run the Automatic Discovery process.
    a. In the **Days** field, click the day tab for each day to run the automatic discovery. Automatic discovery runs on the days that appear dark blue. You must schedule automatic discovery to occur at least once every week. You should run automatic discovery daily at a time when the application is least used.
    b. Click the **Time** drop-downs to enter the automatic discovery start time.
    c. Click the **Time Zone** drop-down and select a time zone.
    d. Click **Discover Now** to run automatic discovery now.
4. Click **Save** to save the General peer settings.

11.2.2.4 Logging

The Logging subtab enables you to manage which user actions are to create log entries. You can view log entries on the Cluster Manager at the appliance level on the System Logs tab. See the Processes and Logs topic for a list of the system logs to where log entries are made.

This subtab enables you to override the cluster level Logging settings for an individual peer. Select the **Override Cluster Setting** check box to enable the following fields.

-  - User actions are logged.
-  - User actions are not logged.

Some user action log functionality is dependent upon your software kernel version being higher than 2.6.36. On the **Administration** > **About** page, click **PHP Status** under **Status Information** to find your kernel version.

- applianceSettingManaged - Cluster Manager Appliance Settings creates log entries when a user changes a setting on the Cluster Manager Appliance Settings tab.
- clusterManaged - Cluster Manager Appliance Management creates log entries when a user performs actions such as database synchronization, fail over, etc. from the gear menu at the appliance level on the Cluster Manager.
- commandExecuted - Console Command Execution creates log entries when a user executes a command in the Linux terminal.
- configFileModified - System Configuration Files creates log entries when various system configuration files are modified.
- devicePluginEntityManaged - Device Editor Plugin Object Managers creates log entries when a device plugin object manager (e.g. "DNS Objects", "ICMP Objects" or "HTTP Objects") is modified.
- devicePluginManaged - Device Editor Plugin Settings creates log entries when a user modifies the plugin settings for a device on the Add/Edit Device page.
- discoveryManaged - Discovery Management creates log entries when a user queues a device discovery, changes discovery priority or cancels discovery.
- entityManaged - General Management triggers when a user creates, updates, deletes, enables or disables devices, alerts, thresholds, policies, users, trap destinations, and others.
- entityMappingManaged - Association Management creates log entries when a user modifies associations of device/object groups, nested device/object groups, user roles, trap destinations or metadata mapping.
- fileUploaded - File Upload Management creates log entries when a file has been uploaded to cluster manager upload update file, status maps or device types.
- importTriggered - Data Import creates log entries when a user imports data via an .spk file.
- processManaged - Cluster Manager Processes creates log entries when a user starts, stops, or restarts a process from the Process Overview tab on the Cluster Manager.
- ruleApplied - Membership Rules triggers when a user applies object group and device group membership rules.
- settingModified - Cluster Manager Settings creates log entries when a user modifies the settings on the **Cluster Settings** tab or the **Peer Settings** tab on the Cluster Manager.
- soapMethodInvoked - SOAP API Call creates log entries when a user invokes a SOAP API call.
- userAuth - User Authentication creates log entries when a user logs in, logs out or is affected by other authentication events such as inactivity time out or failed login attempts.
- userPasswordChanged - User Password creates log entries when a user changes their password or an account is created with a new password.

Click **Save** to save the Logging settings.

11.2.2.5 Poller

The Poller subtab enables you to define poller settings for the peer.

1. Select the **Override Cluster Settings** check box to enable the following field.
2. In the **Poller Threads** field, enter the number of poller threads to use concurrently (between 1 and 1000). The default is 60.

SevOne NMS Appliance Model	Recommended Poller Thread Max
PAS2k	60
PAS5k	60
PAS10k	100
PAS20k	200
PAS60k	300
PAS100k	600
PAS200k	1000

> ⚠ Poller Thread Max should be set to the smallest size of the SevOne NMS appliance model in the cluster. By not doing so, it may result in resource issues.

11.2.2.6 Primary/Secondary

The Primary/Secondary subtab enables you to view the IP addresses for the two appliances that act as one SevOne NMS peer in a Hot Standby Appliance (HSA) peer pair implementation. In a Hot Standby Appliance relationship, the active appliance does the normal network polling and the passive appliance pulls the config database data from the active appliance and pulls the data database data from the active appliance to provide redundancy. The passive appliance takes the active role if the active appliance fails. The primary appliance is initially set up to be the active appliance. If the primary appliance fails, it is still the primary appliance but its role changes to the passive appliance. The secondary appliance is initially set up to be the passive appliance. If the primary appliance fails, the secondary appliance is still the secondary appliance but it becomes the active appliance. You define the appliance IP address upon initial installation. Please refer to *SevOne NMS Installation Guide* for details.

1. In the **Primary Appliance IP Address** field, view the IP address of the primary appliance.
2. In the **Secondary Appliance IP Address** field, view the IP address of the secondary appliance.
3. The **Virtual IP Address** field appears empty unless you implement the primary appliance and the secondary appliance to share a virtual IP address. A virtual IP address is useful when you configure the devices SevOne NMS polls to communicate with a specific appliance IP address because if that appliance fails, the virtual IP address becomes the IP address of what was the passive appliance and the communication from the poller is not blocked because of a different poller IP address.
4. In the **Failover Time** field, enter the number of seconds for the passive appliance to wait for the active appliance to respond before the passive appliance takes over. SevOne NMS pings every 2 seconds and the timeout for a ping is 5 seconds. The default value is 600 seconds. The minimum value is 1 second.

   > ⓘ If you change this setting, you must restart the SevOne Leader / Follower Monitor process for both the active appliance and the passive appliance on the Cluster Manager at the appliance level on the Process Overview tab.

5. Click **Save** to save the Primary/Secondary peer settings.

11.2.2.7 Requestd

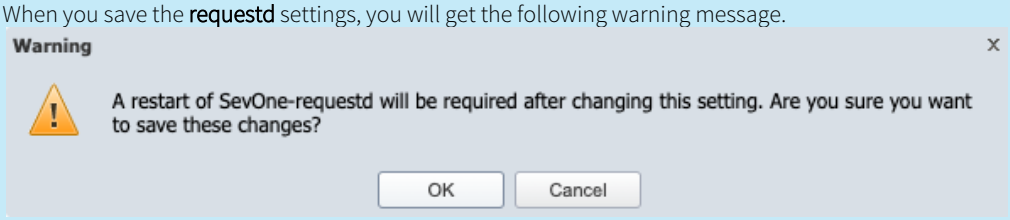
The Requestd subtab allows you to configure **SevOne-requestd** runtime parameters for a peer.

1. Select the **Override Cluster Settings** check box to enable the following fields. It provides local overrides of the **requestd** settings for the selected peer.

   The following settings are provided to support advanced NMS troubleshooting. NMS administrators are strongly discouraged from making changes to these settings without first contacting SevOne Support. Improper changes to these settings may cause service degradation or disruption.

2. In the **Responder Queue Size** field , enter the number of responder tasks to queue up. Maximum number of queries from remote peers that queue up for the local peers to reply to, as the responder threads become available. The default value is 400. The queue size can range between 400 and 1200.
3. In the **Local Threads** field, enter the maximum number of worker threads used for internal **requestd** requests made to the local appliance. The default value is 200. The threads can range between 200 and 600.
4. In the **Originator Threads** field, enter the maximum number of worker threads from the originator. These threads are used for executing the requests from the local appliance (the originator) to the remote appliances. The originator threads are requests that distribute tasks to other peers. The default value is 200. The threads can range between 200 and 600.
5. In the **Responder Threads** field, enter the maximum number of threads used for responding to remote requests from other appliances. The default value is 200. The threads can range between 200 and 600.
6. In the **Requestd Module Originator ZMQ Timeout** field, enter the timeout for the originator ZMQ process that handles the **requestd** queries. 0 minutes indicates no timeout. Timed out queries are discarded, resulting in query failure. Lowering the timeout may help **requestd** from exhausting threads due to excessively long queries or network conditions that may cause ZMQ to wait indefinitely. Setting the value too low may cause reports that are expected to take a long time to run, to timeout or display impartial results. The valid values are 0 minutes (for **no timeout**) or 15 - 1440 minutes. The default value is 0 minutes.
7. Click **Save** to save the Requestd settings.

   When you save the **requestd** settings, you will get the following warning message.

   Click **OK** to save the settings or Cancel to exit.

11.2.2.8 SevOne Data Bus

The SevOne Bus subtab enables you to choose a publisher and restart SevOne Data Bus.

1. Select the **Enable SevOne Data Bus** check box to enable SevOne Data Bus.

   The following fields are only available if SevOne Data Bus is enabled.
   - Select the **Override Publisher** check box to override the default SevOne Data Bus publisher.
   - Click the **Publisher** drop-down list and select the publisher you would like to overwrite and restart.
   - Click **Save** to save the SevOne Data Bus settings before performing a restart.
   - Click **Restart SDB** to restart the SevOne Data Bus.

11.2.2.9 Storage

The Storage subtab enables you to configure storage data retention on an individual peer.

1. Select the **Override Cluster Settings** check box to enable the following field. This subtab enables you to override Data Retention settings for an individual peer.
2. In the **Data Retention** field, enter the number of days' worth of data to store. The default/recommended value is 365 days. Increasing this value means that the physical storage requirements will be much greater. The minimum value is 1 day and the maximum value of 730 days.

   Data retention greater than 365 days is not supported. SevOne recommends you contact SevOne Support before you click Yes to proceed.

3. Click **Save** to save the Storage settings.

11.2.2.10 Syslog

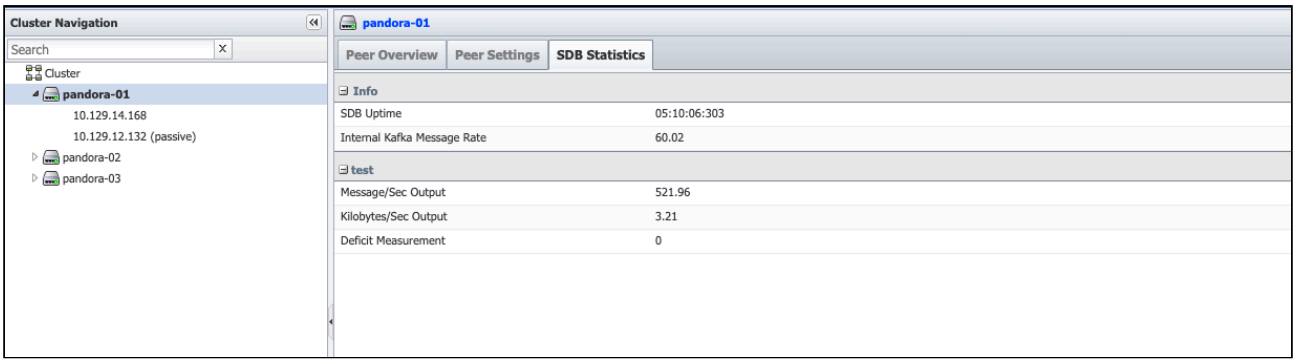
The Syslog subtab enables you to define where this peer is to send Syslog data. This subtab enables you to override the cluster level Syslog destination for an individual peer.

1. Select the **Override Cluster Settings** check box to enable the following fields.
2. Click **Add Syslog Destination** or click to add or edit a Syslog destination.
3. In the **Destination Name** field, enter the name of the host/destination device to which to send the Syslog data.
4. In the **IP Address** field, enter the IP address of the host/destination device.
5. Click the **Protocol** drop-down and select **TCP** or **UDP** or **TLS** for the port type to which to send Syslog data.
6. In the **Port** field, enter the port number to which to send Syslog data.
7. Click **Update** to save the destination.
8. Repeat to add additional destinations to the list.
9. Click **Save** to save the Syslog settings.

> To configure syslog destination(s) using the Command Line Interface, please refer to Configure Syslog Destinations.

11.2.3 SDB Statistics

Click **<peer name>** in the cluster hierarchy on the left and select the **SDB Statistics** tab on the right to view SevOne Data Bus statistics.



SDB Statistics are available only when SevOne Data Bus is **active** and **running** otherwise, no statistics are available. Provides you with key performance indicators (KPIs) such as,

- **SDB Uptime**
- **Internal Kafka Message Rate** *(per second)* - generated by SevOne Data Bus. This is the number of messages seen since the SDB process started, divided by the amount of time SDB has been running.
- **Message/Sec Output** *(for each publisher)* - generated by SevOne Data Bus. This is the number of messages sent to the external broker since the SDB process started, divided by the amount of time SDB has been running.
- **Kilobytes/Sec Output** *(for each publisher)* - generated by the broker client library used by SevOne Data Bus. The method of calculation is unknown.
- **Deficit Measurement** *(for each publisher)* - generated by SevOne Data Bus. This is the number of messages sent to the external broker, less the number of successful reported back by the external broker.

11.3 Appliance Level - Appliance Overview, Appliance Settings, System Settings, Process Overview, System Logs, Integration, Appliance License

<peer name> - Click the triangle next to the peer level icon in the hierarchy to display the IP address of the appliance that makes up the peer.
For a Hot Standby Appliance peer pair implementation two appliances appear.

- The primary appliance appears first in the peer pair.
- The secondary appliance appears second in the peer pair.
- The active appliance that is actively polling does not display any additional indicators.

- The passive appliance in the peer pair displays *(passive)*.

<IP address> - When you click on an appliance IP address in the cluster hierarchy on the left, the following tabs appear on the right to enable you to view appliance level information and to define appliance level settings.

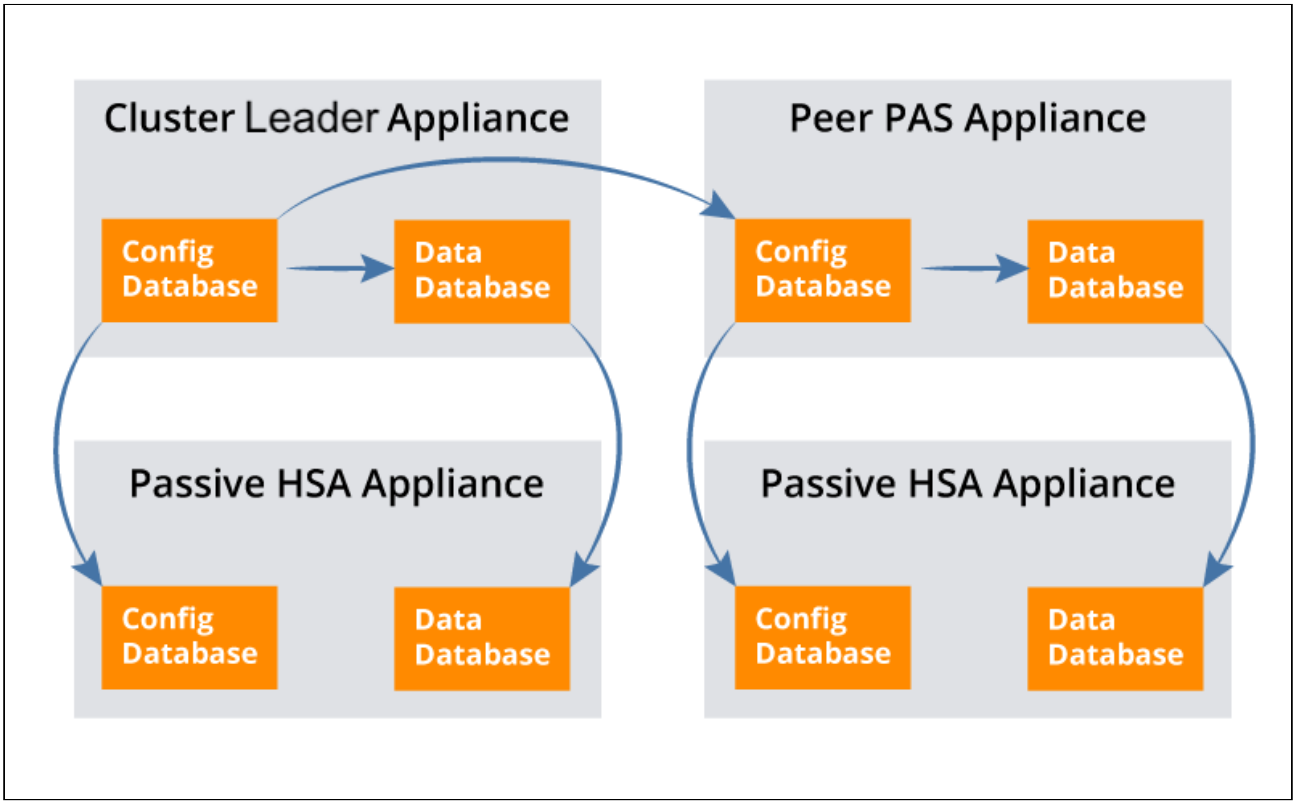
- **Appliance Overview** - Enables you to view appliance level information including the status of the replication of the SevOne NMS databases. See below for details.
- **Appliance Settings** - Enables you to make the appliance conform to Common Criteria security standards.
- **System Settings** - Enables you to read/write the various **SevOne-select** settings, available from the Command Line Interface, the appliance is using.
- **Process Overview** - Enables you to view the list of processes SevOne NMS runs.
- **System Logs** - Enables you to view the data SevOne NMS writes to log files.
- **Integration** - Enables you to add a new appliance to your cluster as a new peer. If you plan to add a new appliance to your cluster as a Hot Standby Appliance you must contact SevOne Support.
- **Appliance License** - Enables you to view SevOne NMS details for the appliance you are logged into.

11.3.1 Database Replication Explanation

The SevOne NMS application peer-to-peer architecture has two fundamental databases.

Config Database - The config database stores configuration settings such as cluster settings, security settings, device settings, etc. SevOne NMS saves the configuration settings you define (on any peer in the cluster) in the config database on cluster leader peer. All active appliances in the cluster pull config database changes from the cluster leader peer config database. Each passive appliance in a Hot Standby Appliance (HSA) peer pair pulls its active appliance's config database to replicate the config database onto the passive appliance.

Data Database - The data database stores a copy of the config database plus all poll data for the devices/objects that the peer polls. The config database on an active appliance replicates to the data database on the appliance. Each passive appliance in a Hot Standby Appliance peer pair pulls its active appliance's data database to replicate the data database onto the passive appliance.



11.3.2 Appliance Level Actions

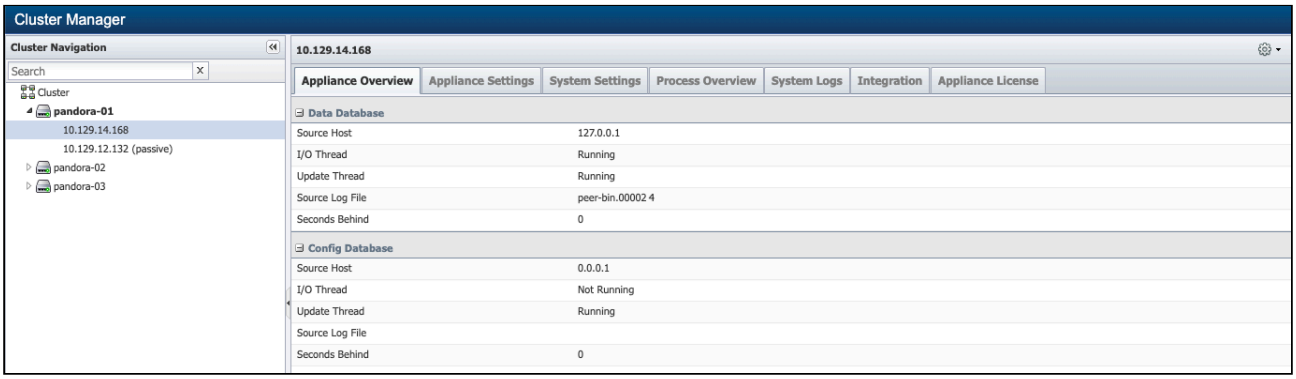
A  appears above the right side on the Cluster Manager to perform appliance level actions. The options that appear are dependent on the appliance you select in the hierarchy on the left side.

Click  and select the following options.

- Select **Device Summary** to display a link to the Device Summary and links to the report templates that are applicable for the device. For details, please refer to section **Device Summary** in *SevOne NMS User Guide*.
- Select **Fail Over** to have the active appliance in a Hot Standby Appliance peer pair become the passive appliance in the peer pair. This option appears when you select the active appliance in a Hot Standby Appliance peer pair.
- Select **Take Over** to have the passive appliance in a Hot Standby Appliance peer pair become the active appliance in the in the peer pair. This option appears when you select the passive appliance in a Hot Standby Appliance peer pair.
- Select **Resynchronize Data Database** to have an active appliance pull the data from its own config database to its data database or to have the passive appliance in a Hot Standby Appliance peer pair pull the data from the active appliance's data database.
- Select **Resynchronize Config Database** to have an active appliance pull the data from the cluster leader peer's config database to the active peer's config database or to have the passive appliance in a Hot Standby Appliance peer pair pull the data from the active appliance's config database.
- Select **Rectify Split Brain** to rectify situations when both appliances in a Hot Standby Appliance peer pair think they are active or both appliances think they are passive. Both appliances in a Hot Standby Appliance peer pair can end up in an active state when the Internet connection between the appliances is interrupted.
    - When you logged on, you received an administrative message that stated either "Neither appliance in your Hot Standby Appliance peer pair with IP addresses <n> and <n> is in an active state." or "Both appliances in your Hot Standby Appliance peer pair with IP addresses <n> and <n> are either active or both appliances are passive."
    - When you select one of the affected appliances in the hierarchy on the left side of the Cluster Manager this option appears.
    - When both appliances think they are passive and you select this option, the appliance for which you select this option becomes the active appliance in the Hot Standby Appliance peer pair.
    - When both appliances think they are active and you select this option, the appliance for which you select this option becomes the passive appliance in the Hot Standby Appliance peer pair.

11.3.3 Appliance Overview

Click  next to a peer in the cluster hierarchy on the left side, click **<appliance IP address>**, and then select the **Appliance Overview** tab on the right to display appliance level information.



Data Database Information

- **Source Host** - Displays the IP address of the source from where the appliance replicates the data database. In a single appliance implementation and on an active appliance, this is the IP address of the appliance itself. HSA passive appliance data database replicates from the active appliance data database.
- **I/O Thread** - Displays *Running* when an active appliance is querying its config database for updates for the data database. Displays *Not Running* when the appliance is not querying the config database. HSA passive appliance data database queries the active appliance data database.
- **Update Thread** - Displays *Running* when the appliance is in the process of replicating the config database to the data database. Displays *Not Running* when the appliance is not currently replicating to the data database.
- **Source Log File** - Displays the name of the log file the appliance reads to determine if it needs to replicate the config database to the data database.
- **Seconds Behind** - Displays 0 (zero) when the data database is in sync with the config database or displays the number of seconds that synchronization is behind.

Config Database Information

- **Source Host** - Displays the IP address of the source from where the appliance replicates the config database. In a single appliance implementation and on the cluster leader peer active appliance, this is the IP address of the appliance itself. HSA passive appliance config database replicates from the active appliance config database.
- **I/O Thread** - Displays *Running* when an active appliance is querying the cluster leader peer config database for updates. Displays *Not Running* when the appliance is not querying the cluster leader peer config database. HSA passive appliance config database queries the active appliance config database.
- **Update Thread** - Displays *Running* when the appliance is in the process of replicating the config database. Displays *Not Running* when the appliance is not replicating the config database.
- **Source Log File** - Displays the name of the log file the appliance reads to determine if it needs to replicate the config database.
- **Seconds Behind** - Displays 0 (zero) when the config database is in sync with the cluster leader peer config database or displays the number of seconds that the synchronization is behind.

11.3.4 Appliance Settings

Click ▷ next to a peer in the cluster hierarchy on the left, click **<appliance IP address>**, and then select the Appliance Settings tab on the right side to define the settings that enable the appliance to meet Common Criteria security standards.

11.3.4.1 Common Criteria

Prerequisites:

- The peer cannot be a part of a cluster.
- The peer cannot have a Hot Standby Appliance.
- You must log on to the appliance via HTTPS.
- xStats adapter configuration is not available.
- Group aggregated indicator features are not available.

Perform the following steps to enable the appliance to meet Common Criteria security standards.

1. Select the **Enable Common Criteria** check box.
2. Click **Save** to display a confirmation message pop-up.
3. Click **OK** on the pop-up to display another confirmation pop-up that informs you that a restart is required to enable Common Criteria mode.
4. Click **OK** on the second confirmation pop-up to start the Common Criteria enable process and to restart the appliance. If you click Cancel, the common Criteria enable process starts but remains incomplete until after the appliance is restarted.
5. Watch the status messages as the system checks and adjusts settings to meet Common Criteria standards. The page displays nine green check marks to display the success of the Common Criteria mode success.

   > ⓘ If you did not click **OK** to restart the appliance, you must restart the appliance before the Common Criteria mode is enabled.

6. Click **Save**. A Date and Time subtab appears to enable you to define the appliance system date and time for Common Criteria.

11.3.4.2 Date and Time

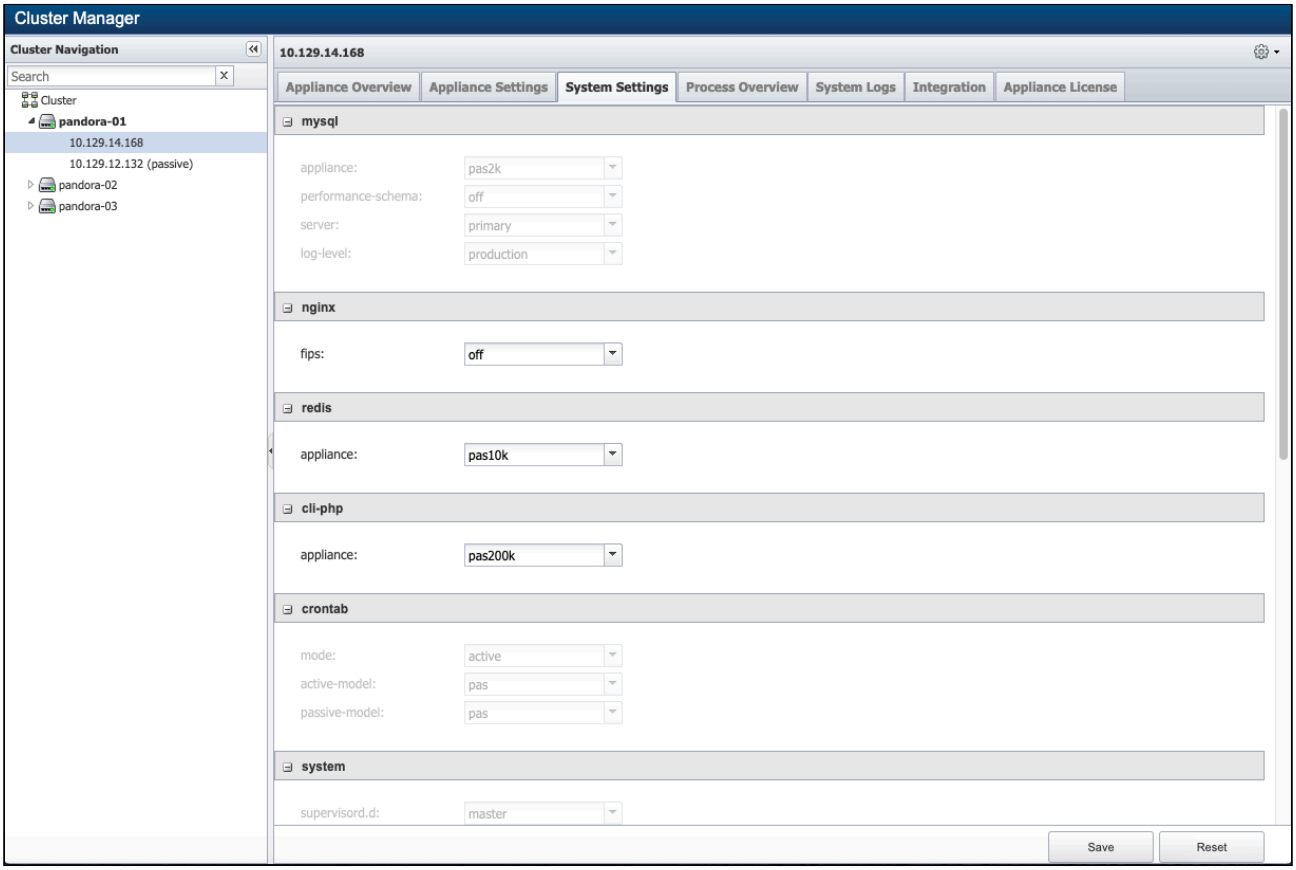
The Appliance Settings tab displays a Date and Time subtab when you implement the Common Criteria mode to enable you to define the system time for the appliance.

1. In the **Date and Time** field, enter the system time for the appliance.
2. Click **Save** to save the Date and Time settings.

11.3.5 System Settings

> ⚠ Kernel does not support the FIPS Mode.

Click ▷ next to a peer in the cluster hierarchy on the left side, click **<appliance IP address>**, and then select the **System Settings** tab on the right to read/write the various **SevOne-select** settings, available from the Command Line Interface, the appliance is using.



The following modules are available.

1. mysql - Configure the MySQL config files.
    a. appliance: Configure the appliance type. Value can be pas2k, pas10k, pas20k, pas40k, pas60k, pas200k, or pas300k.
    b. performance-schema: Toggle the PERFORMANCE SCHEMA (optional). Value can be on or off.
    c. server: Configure whether or not this is the Primary or Secondary appliance. Value can be primary or secondary.
    d. log-level: Configure whether or not to log warnings to the MySQL error logs. Value can be debug or production.
2. nginx - Configure the nginx config files. NOTE: When nginx setting is changed, it will cause the server to reboot and a refresh will be required.
    a. fips: Configure nginx to be fips compliant. Value can be on or off.
3. redis - Configure the Redis config files.
    a. appliance: Configure the appliance type. Value can be pas10k, pas20k, pas40k, pas60k, pas200k, or pas300k.
4. cli-php - Configure the CLI PHP config files.
    a. appliance: Configure the appliance type. Value can be pas10k, pas20k, pas40k, pas60k, pas200k, or pas300k.
5. crontab - Configure the crontab files.
    a. mode: Configure the 'mode' to use. Value can be active or passive.
    b. active-model: Configure the active model to use. Value can be dnc or pas.
    c. passive-model: Configure the passive model to use. Value can be dnc or pas.
6. system - Configure the system files.
    a. supervisord.d: Configure the 'supervisord.d' directory to use. Value can be dnc, fips, master, or slave.
7. openldap - Configure the ldap files.
    a. ldap.conf: Configure the LDAP config to use. Value can be cert or nocert.
8. ssh - Configure the ssh config files. NOTE: This module is disabled because changing ssh mode will result in the failure of the current ssh cipher.
    a. config: Configure ssh to be fips compliant. Value can be fips or default.
9. sshd - Configure the sshd config files. NOTE: This module is disabled because changing sshd mode will result in the failure of the current ssh cipher.
    a. config: Configure sshd to be fips compliant. Value can be fips or default.
10. kafka - Configure the kafka config files.
    a. appliance: Configure the appliance type. Value can be dnc, pas2k, pas4k, pas10k, pas20k, pas40k, pas60k, pas200k, or pas300k.
11. zookeeper - Configure the kafka config files.

    a. **appliance**: Configure the appliance type. Value can be **dnc**, **pas2k**, **pas4k**, **pas10k**, **pas20k**, **pas40k**, **pas60k**, **pas200k**, or **pas300k**.
12. **data-bus** - Configure the data bus config files.
    a. **appliance**: Configure the appliance type. Value can be **pas2k**, **pas10k**, **pas20k**, **pas40k**, **pas60k**, **pas200k**, **pas300k**, or **disabled**.
13. **php-fpm** - Configure the PHP-FPM config files. **NOTE**: Changing the mode of php-fpm will cause the server to restart. You are required to refresh the page after changing this value.
    a. **process-manager**: Configure the Pool Process Manager (pm). Value can be **dnc100**, **dnc1000**, **dnc1000hf**, **dnc1500**, **dnc1500hf**, **dnc200**, **dnc400**, **dnc600**, **pas5k**, **pas10k**, **pas20k**, **pas40k**, **pas60k**, **pas200k**, or **pas300k**.

Click on **Reset** button to set the values to current settings.
Click on **Save** button to apply the changes.

11.3.6 Process Overview

Click ▷ next to a peer in the cluster hierarchy on the left, click **<appliance IP address>**, and then select the **Process Overview** tab on the right side to display a list of processes.

-  - Click to refresh the process information or to refresh the information at the frequency you select.
- **Shutdown Appliance** - Click to shut down the appliance
- **Restart Appliance** - Click to restart the appliance.

Processes appear grouped in subsections. Process information includes the process name, the path to the process file, the number of instances of the process, the percentage of CPU the process is using, and the amount of RAM the process uses.

Stop, **Start**, and **Restart** buttons enable you to stop and start some processes. You should not click these buttons without strong cause.

See the Processes and Logs chapter for a list of processes.

11.3.7 System Logs

Click ▷ next to a peer in the cluster hierarchy on the left, click **<appliance IP address>**, and then select the **System Logs** tab to view appliance level logs. SevOne NMS is a Linux application with various daemons and background utilities that run at all times. Most of these record their activities in logs on the appliance.

The upper section of the tab enables you to select the log to view. Log data refreshes upon each selection from the drop-down menus. Logs display the newest data at the bottom. When you view a log, the display scrolls to the bottom of the log.

Please refer to Processes and Logs chapter for a list of log files.

1. Click the **Select log...** drop-down and select the log to view.
2. Click the **Last <n> Lines** drop-down and select how many lines at the end of the log file to display.
3. Click **Download Full Log File** to export the log to a .log file.
4. Click **Refresh** to update the System Logs display.

11.3.8 Integration

For a new appliance or when you want to move a peer to a different SevOne NMS cluster in a multi-cluster environment, the Integration tab enables you to add **this** appliance as a new peer to your SevOne NMS cluster or to move **this** peer to a different SevOne NMS cluster in your network when you have a multi-cluster environment.

From the Cluster Manager, click ▷ in the cluster hierarchy on the left side next to the peer to add/move to display the peer's IP address. Click on the IP address and then select the **Integration** tab on the right side.

⚠ Please Note:

- All data on this appliance will be deleted.
- You need the name of this appliance.
- You need the IP address of this appliance.

- You need to be able to access the Cluster Manager on a SevOne NMS peer that is already in the cluster to which you intend to add this appliance.
- If you do not complete the steps within ten minutes, you must start again at step 1) Click **Allow Peering**... to queue this appliance for peering within the following ten minutes.

After you click **Allow Peering** on this tab, you will have ten minutes to perform the following steps from a peer that is already in the cluster to which you intend to add this peer/appliance.

1. Click **Allow Peering** here on this tab.

   This invokes a pre-health check to be performed. Only if the pre-health check completes successfully, you are allowed to proceed to the next step. The pre-health checks include:
   - **Services** - ensures all required services are running on the peer. For example, Kafka, mysqld, REST API, etc.
   - **Ports** - all required ports are open. For example, TCP 22, TCP 443, TCP 3306, TCP 3307, TCP 9092, TCP 60007, etc.

   The following are some examples of possible pre-health check failure error messages.
   - ERROR_PORTS_CLOSED, port check failed.
   - ERROR_SERVICES_UNHEALTHY, service check failed.
   - ERROR_SSH_FAILED, ssh check failed.
2. Log on to the peer in the destination cluster.
3. Go to the Cluster Manager (Administration > Cluster Manager).
4. At the  Cluster level select the **Peers** tab.
5. Click **Add Peer** to display a pop-up.
6. Enter the **Peer Name** and the **IP Address** of this peer/appliance.
7. On the pop-up, click **Add Peer**.
   - **All data on the peer/appliance you are adding is deleted.**
   - Do not do anything on the peer you are adding until a **Success** message appears on the peer on which you click Add Peer.
   - You can continue working and performing business as usual on all peers that are already in the cluster.
   - The new peer appears on the Peers tab in the destination cluster with a status message. Click  to update the status.
   - The new peer appears in the cluster hierarchy on the left.

   At this point, pre-health check is performed on the Cluster Leader to which the peer is being added to. You are allowed to proceed to the next step only if the pre-health check completes successfully. The pre-health check include:
   - **Services** - ensures all required services are running on the peer. For example, Kafka, mysqld, REST API, etc.
   - **Ports** - ensures that all the required ports open on the Cluster Leader are also open on the target peer. For example, TCP 22, TCP 443, TCP 3306, TCP 3307, TCP 9092, TCP 60007, etc.
   - **SSH** - ensures that Cluster Leader can access or communicate with the target peer.

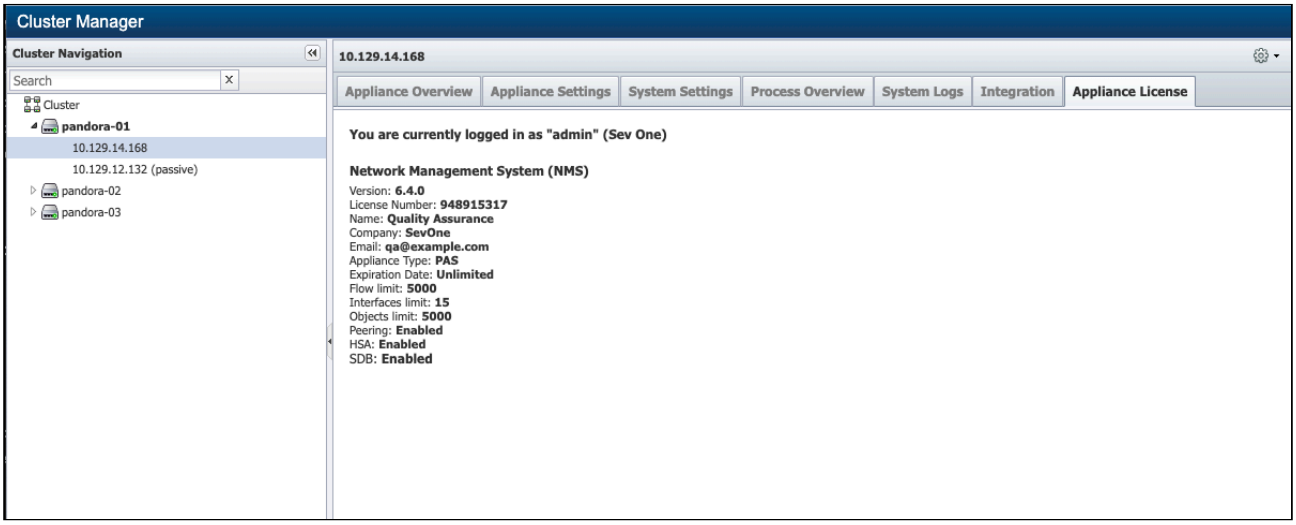
   The following are some examples of possible pre-health check failure error messages.
   - ERROR_PORTS_CLOSED, port check failed.
   - ERROR_SERVICES_UNHEALTHY, service check failed.
   - ERROR_SSH_FAILED, ssh check failed.
8. After the Success message appears on the peer in the destination cluster, you can go to the peer you just added and the entire cluster hierarchy to which you added the peer should appear on the left in the Cluster Manager.
9. You can use the Device Mover to move devices to the new peer.

If the integration fails, click **View Cluster Logs** on the Peers tab on the peer that is in the destination cluster to display a log of the integration messages.

Click **Clear Failed** to remove failed attempts from the list. Failed attempts are not automatically removed from the list which enables you to navigate away from the Peers tab during the integration.

11.3.9 Appliance License

Provides the following SevOne NMS details for the appliance you are logged into.



When a peer exceeds the object capacity, that peer does not discover any objects that go beyond the peer capacity. This prevents a peer from being overloaded which impacts the integrity of the peer. No metrics are collected from objects that are not discovered.
Admin receives the following message at login:
Peer <peer name> is at <n> capacity.
This message indicates that a peer in your cluster exceeds its object capacity. A peer does not discover any new devices or poll additional objects when a peer reaches its object capacity.

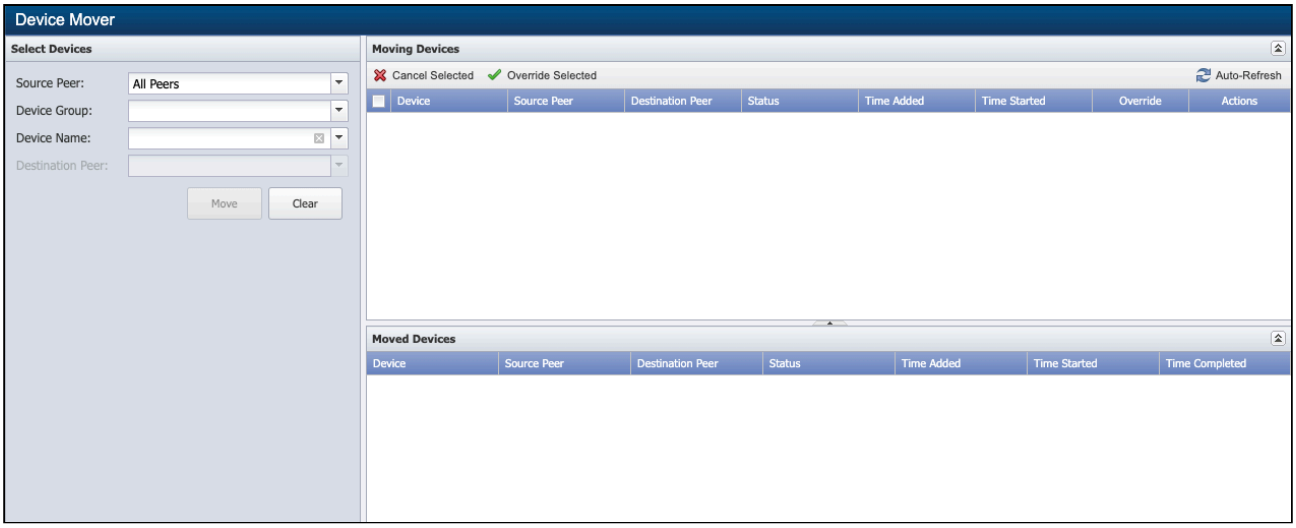
As of SevOne NMS 6.1 or above releases, license file is no longer needed.

12 Device Mover

The Device Mover enables you to move a device from one peer to another peer when you have a multi-peer cluster.

To access the Device Mover from the navigation bar, click the **Devices** menu and select **Device Mover**. You can also access the Device Mover from the Edit Device page.

If you have a single peer cluster, the fields on the Device Mover are disabled.



There are several reasons to move devices from one peer to another.

- When a peer exceeds the license capacity, that peer does not discover any objects that go beyond the peer capacity. This prevents a peer from being overloaded which impacts the integrity of the peer. No metrics are collected from objects that are not discovered.
- When a peer discovers a device and the device is physically closer to a different peer, you may want to move the device to the peer that is physically closer to prevent latency issues.

12.1 Move Devices

The left side enables you to select the source peer from which to move devices, the devices to move, and the destination peer to which to move the devices. Devices that have a large number of polled objects can take a relatively long time period to move because of the amount of data collected for the device.

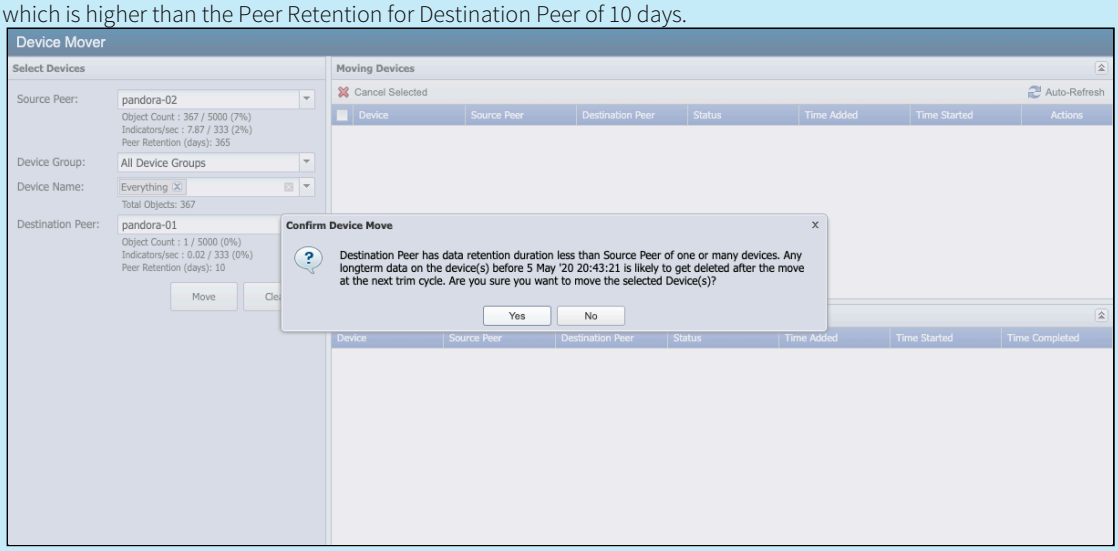
1. Click the **Source Peer** drop-down and select the peer from which you want to move devices.
    - Select **All Peers** to display all devices in the Device Name drop-down list. This enables you to move any device from any peer.
    - Select a specific peer to display the devices the peer polls in the Device Name drop-down list. When you select a peer, you see the following details under the Source Peer field.
        - **Object Count** - displays the number of objects the peer uses, the number of objects the peer is licensed to use, and the percentage of the license capacity the peer uses. The value in this field is derived from **Administration** > **Cluster Manager/Peers** display.
        - **Indicators/sec** - displays the total number of indicators the peer receives per second from all interfaces. The value in this field is derived from **Administration** > **Cluster Manager/Peers** display.
        - **Peer Retention (days)** - data retention in days from the source peer. The value in this field is derived from **Administration** > **Cluster Manager/Peers** display.
2. Click the **Device Group** drop-down and select a device group/device type to display devices that are members of the device group/device type in the Device Name drop-down list.
3. Click the **Device Name** drop-down and select the devices to move. The number of objects polled on the selected devices displays below the Device Name field.
4. Click the **Destination Peer** drop-down and select the peer to which to move the devices you select. The capacity of the peer you select appears below the Destination Peer field.
    - **Object Count** - displays the number of objects the peer uses, the number of objects the peer is licensed to use, and the percentage of the license capacity the peer uses. The value in this field is derived from **Administration** > **Cluster Manager/Peers** display.

- **Indicators/sec** - displays the total number of indicators the peer receives per second from all interfaces. The value in this field is derived from **Administration** > **Cluster Manager/Peers** display.
- **Peer Retention (days)** - data retention in days from the destination peer. The value in this field is derived from **Administration** > **Cluster Manager/Peers** display.

5. Click **Move** to add the devices to the Moving Devices section on the right. Devices in the Moving Devices section are queued to move the next time the move engine runs.

Move a device from a peer with a higher retention value to a peer with lower retention value

If a device is moved from a peer with a **higher** retention value to a peer with a **lower** retention value, you will get a warning message to confirm the device move. In the example below, Peer Retention for Source Peer is 365 days which is higher than the Peer Retention for Destination Peer of 10 days.

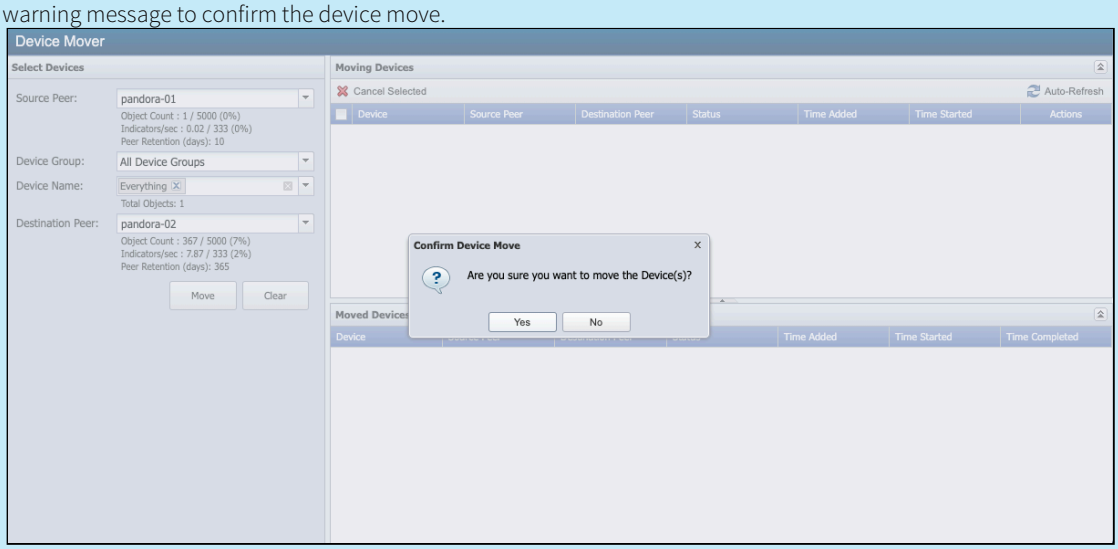


When you click on **Yes**, the device is queued for the move. If you click on **No**, you have the option to Clear or make any updates to the values in the fields.

When the data retention duration of the Destination Peer is less than the duration of the Source Peer, you will lose data on the destination.

Move a device from a peer with a lower retention value to a peer with higher retention value

If a device is moved from a peer with a **lower** retention value to a peer with a **higher** retention value, you will get a warning message to confirm the device move.



When you click on **Yes**, the device is queued for the move. If you click on **No**, you have the option to Clear or make any updates to the values in the fields.

12.1.1 Moving Devices

The Moving Devices section displays the list of devices you schedule to move and enables you to monitor the progress of the move.

- Select the check box for each device you want to cancel the move and click **Cancel Selected** or click X in the Actions column. After a move starts you cannot cancel the move. Successful moves are removed from the list.
- **Device** - Displays the name of the device to move. Click the device name to display the Edit Device page for the device.
- **Source Peer** - Displays the name of the peer from which the device is to move. Click the peer name to display the Cluster Manager appliance level statistics for the source peer.
- **Destination Peer** - Displays the name of the peer to which the device is to move. Click the peer name to display the *Cluster* Manager appliance level statistics for the destination peer.
- **Status** - Displays the status of the move. Displays *n/a* when the move has yet to occur.
- **Time Added** - Displays the time the move was added to the list.
- **Time Started** - Displays the time the move started.
- **Time Completed** - Displays the time the move was completed.
- **Override** - Select the check box(es) for the device(s) that have failed to move due to connectivity issues. By doing this, the check boxes for the devices selected will override the connectivity check.

12.1.2 Moved Devices

The Moved Devices section displays the list of devices that have been moved.

- **Device** - Displays the name of the device that was moved. Click the device name to display the Edit Device page for the device.
- **Source Peer** - Displays the name of the peer from which the device was moved. Click the peer name to display the Cluster Manager appliance level statistics for the source peer.
- **Destination Peer** - Displays the name of the peer to which the device was moved. Click the peer name to display the Cluster Manager appliance level statistics for the destination peer.
- **Status** - Displays the status of the move.
- **Time Added** - Displays the time the move was added to the list.
- **Time Started** - Displays the time the move started.
- **Time Completed** - Displays the time the move was completed.

12.2 Flow Falcon Device Mover

12.2.1 Pre-Checks

Prior to Move NetFlow Devices, execute the pre-checks to ensure that the destination peer has the resources to handle the devices moved from the source peer to the destination peer. Option, **-p, --do-pre-checks**, performs all pre-checks. There are other options such as **-c, --only-check-capacity** (existing option) or **-d, --only-check-disk-space** (new option), which allow you to perform the checks one by one.

> The checks must be run on the **source peer**.

Example

Run NetFlow Device Mover with Pre-Checks option

```
$ SevOne-act flowdb move -p --label devMove1 --device 10.2.12.199 --remote-peer-ip 10.129.13.66 --
verbose
```

The checks stop on the first error. If you move more than one device, the listed error will not represent all the resources required for them, unless the error is on the last device in the list.
To check disk space, use option **-d**.
To check the capacity (number of Flow interfaces), use option **-c**.

After executing the **SevOne-act flowdb move** command above, logs can be found in **/var/SevOne/flowdb-move.log**.

Examples of errors (without --verbose option)

Capacity check error

2019-06-24T14:15:48+00:00 Checking device: 172.24.0.195
2019-06-24T14:15:48+00:00 Interface Limit on remote machine is 300. It already has 155 and you are trying to add (at least) 150 new interfaces.
2019-06-24T14:15:48+00:00 01. Check remote peer capacity.
2019-06-24T14:15:48+00:00 [FAIL]
2019-06-24T14:15:48+00:00 Step 01. Check remote peer capacity. FAILED. Exiting Netflow device mover.

Disk Space check error

2019-06-24T14:15:48+00:00 Checking device: 172.24.0.195
2019-06-24T14:15:48+00:00 01. Check remote peer capacity.
2019-06-24T14:15:48+00:00 [OK]
2019-06-24T14:15:48+00:00 Available disk space on remote peer is 123456789. It is not enough! Total needed disk space is (at least) 129456789.
2019-06-24T14:15:48+00:00 02. Check remote peer disk space.
2019-06-24T14:15:48+00:00 [FAIL]
2019-06-24T14:15:48+00:00 Step 02. Check remote peer disk space. FAILED. Exiting Netflow device mover.

Example: Successful check

2019-06-24T14:15:48+00:00 Checking device: 172.24.0.187
2019-06-24T14:15:48+00:00 01. Check remote peer capacity.
2019-06-24T14:15:48+00:00 [OK]
2019-06-24T14:15:48+00:00 02. Check remote peer disk space.
2019-06-24T14:15:48+00:00 [OK]

12.2.2 Move NetFlow Devices

NetFlow device mover switches the flow traffic from one SevOne Dedicated NetFlow Collector to another (**DNC-1 to DNC-2** or **NMS to DNC** or **NMS to NMS** or **DNC to NMS**) and then, backfills the missing data. This ensures minimal data loss since the downtime is only the amount of time it takes you to switch the flow traffic and not the transfer time itself. Prior to moving the device, you must execute Pre-Checks to ensure that you have enough disk space and capacity on the device you are moving to, to perform the move.

While NetFlow device move operation is in progress anywhere in the cluster, the NetFlow discovery is locked on the **source** and **destination** pair only. When discovery is locked, on the source and destination pair,

- No new Netflow devices are discovered.
- No changes to existing Netflow devices are identified i.e., addition of new interfaces.

After device move operation completes, discovery runs again and discovers the new flows which were blocked during the device move. Existing flows continue to be collected from all existing devices without any impact to flow collection and reporting.

During the NetFlow device move operation, there is no capability to create any FlowFalcon Reports for the devices that are currently being moved until the device move operation has completed successfully and the NetFlow discovery process has executed on the **destination peer** of the NetFlow device move operation. All other NetFlow devices that are not undergoing a NetFlow device move operation continue to report normally using FlowFalcon Reports.

Example: Steps to move NetFlow devices from DNC-1 to DNC-2

1. Stop NetFlow for the device you are moving on **DNC-1**.
   You can either stop the NetFlow or create NetFlow firewall rules using Flow Interface Manager to block the incoming traffic for that device.
2. Start NetFlow for the device you are moving on **DNC-2**.
   Make sure you have stopped flow to DNC-1 first before starting flow on DNC-2. Otherwise, you will have duplicate data on both DNCs.
3. If **DNC-2** has global **Deny All** rules set, run migration script **/usr/local/scripts/utilities/update-netflow-firewall-permit.sh** on **DNC-1** to enable migration of NetFlow permissions from **DNC-1** to **DNC-2**.

   ```
   $ /usr/local/scripts/utilities/update-netflow-firewall-permit.sh --remote-peer-ip|-r <Remote peer IP> --device-file|-f <File with Device IP list>
   ```

   - Both, **remote-peer-ip** and **device-file**, are mandatory options.
   - Current version of update-netflow-firewall-permit.sh requires you to provide option **device-file** with a list of devices even if there is only one device in the list.
   - update-netflow-firewall-permit.sh must be executed on the **source peer** prior to moving the device.
4. Check run-time of **ffupdater**.

   ```
   $ grep -r ffupdater /etc/cron-available.d/
   ```
5. Before a device is moved, the device move process checks to see if the **ffupdater** cycle, in progress, has completed successfully as the flow is redirected. If the **ffupdater** cycle has not yet completed, it will internally wait for the cycle to complete. Once the cycle completes, it will automatically process the device move.

   To force a device move, flag **--force-move** has been added. You will get a message informing you of the possible consequences for using the flag.
6. Execute **SevOne-act flowdb move** command on **DNC-1** for your particular device or give it a list, to move multiple devices.

When you execute the **SevOne-act flowdb move** command, you must be in a **screen** session to ensure that the move operations complete successfully and do not abort due to **SSH** connection timeout.

Start screen session

```
$ screen
```

If you are only performing a check using the **-p**, **-c**, **-d** options with **SevOne-act flowdb move** command, you are not required to be in a screen session.

a. You need to specify label for the move. If something goes wrong, using the label, you can check the logs or redo the move.
b. You must specify the device IP address you want to move and the peer IP address you want to move it to.

```
$ SevOne-act flowdb move --label devMove1 --device 10.2.12.199 --remote-peer-ip
10.129.13.66 --verbose
```

c. To move multiple devices, you must have a list of devices in a file (each device IP address must be on a new row). Execute the following command to make the move.

```
$ SevOne-act flowdb move --label devMove1 --device-file myDeviceFile.txt --remote-peer-
ip 10.129.13.66 --verbose
```

- label must always be less than 10 characters long.
- There must be **no** blank lines in the device file.

After executing the **SevOne-act flowdb move** command above, logs can be found in **/var/SevOne/flowdb-move.log**.

7. Restart **SevOne-flowdbd** on DNC-1. This can cause data loss for 1 minute. Each minute, Netflowd sends raw data to SevOne-flowdb. If SevOne-flowdb is in the middle of processing the raw data, data loss may occur. Data loss may also occur in Aggregated Data during the period of move process.
   a. You must restart **SevOne-flowdbd** to update its internal state about the missing data which has been transferred to DNC-2.
   b. If you are planning to decommission DNC-1, you can skip the following step.

```
$ supervisorctl restart SevOne-flowdbd
```

All available parameters can be seen using --help.

Available parameters (--help)

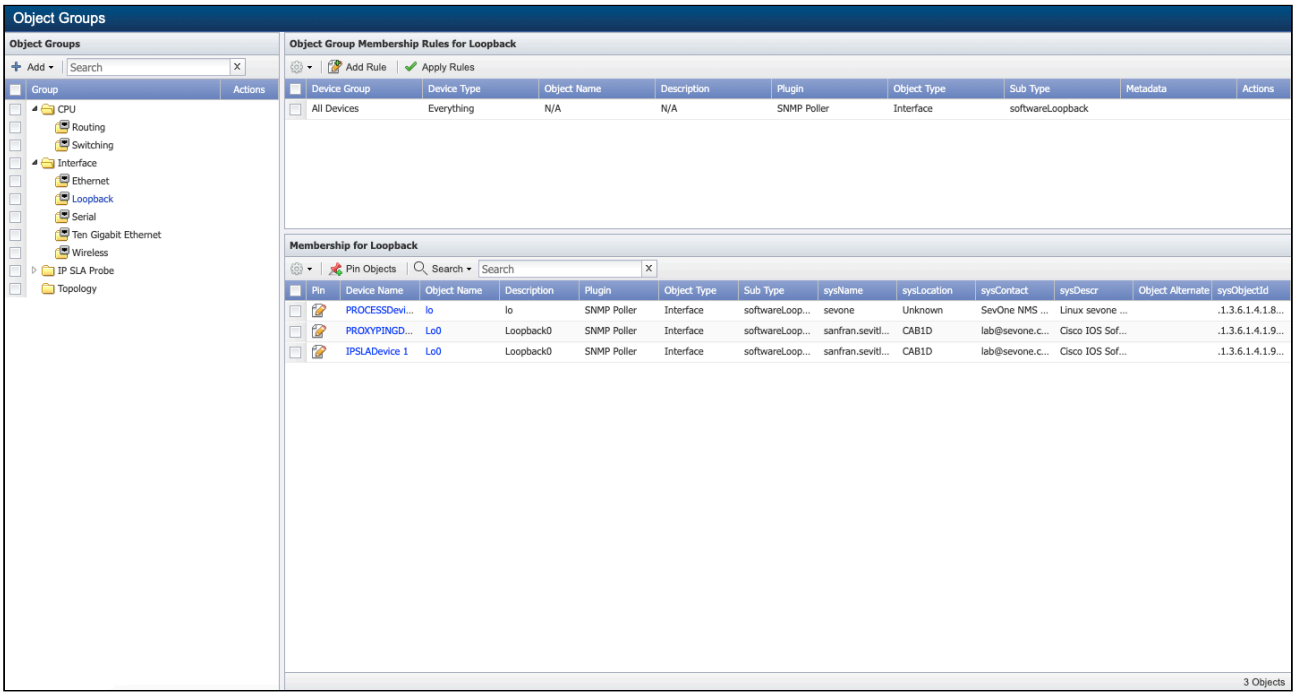
```
$ SevOne-act flowdb move --help
Usage:
$   [ OPTIONS ]

    Move Netflow device to other peer.
```

```
This script accepts the following options:
Flags                   Description
================================================================================
--remote-peer-ip        (Required) Remote peer IP.
--device                (Optional) Device IP.
                        Default:

--device-file           (Optional) File with Device IP list.
                        Default:

--label                 (Required) Label.
--add-iptable-rules     (Optional)
--remove-iptable-rules  (Optional)
-c, --only-check-capacity(Optional)
-d, --only-check-disk-space(Optional)
-p, --do-pre-checks     (Optional)
-h, --help              (Optional) Print this help message, more info with
                        --verbose
-q, --quiet             (Optional) Hide all of the logging output
-v, --verbose           (Optional) Shows all of the logging output, used for
                        debugging
--no-color              (Optional) Pass this flag to disable the use of color.
--serialize-as-json     (Optional) Pass this flag to have the output serialized
                        as json at the command line
--lock                  (Optional) Use this to force a process lock
--conf-file             (Optional) Use this setting to read settings in from a
                        conf file
--export-conf-file      (Optional) Use this setting to write the current setting
                        to a conf file.  The script will still execute
```

13 Object Groups

The Object Groups page enables you to manage the object groups that segment the enabled, visible objects for reports and alerts. You should outline your object groups to best suit your report requirements. Object groups have no effect on how objects are stored. If you plan your implementation appropriately, object group membership rules enable you to automatically assign objects to object groups. You can manually pin (add) objects to object groups.

To access the Object Groups page from the navigation bar, click the **Devices** menu, select **Grouping**, and then select **Object Groups**.



13.1 Object Group Hierarchy

The object group hierarchy appears on the left. The hierarchy is two levels; object classes and object groups. First level object classes enable you to organize the second level object groups. Each object class is a logical group of object groups. You pin (add) objects into the second level object groups. Check boxes enable you to simultaneously manage the metadata values, drag and drop, and delete multiple object groups.

1. - Click to expand the object group hierarchy.
2. Click **Add**.
   - Select **Class** to add a first level object class under which you can add object groups.
   - Select **Group** to add a second level object group into which you can add objects.
3. - Click to edit the values for the metadata attributes you want to associate with the object group. See the Manage Metadata Values section below.
4. - Click to edit the object class name or the object group name.
5. Drag and drop object groups to different classes in the hierarchy.

13.1.1 Manage Metadata Values

The in the object group hierarchy Actions column provides access to the Edit Metadata pop-up that enables you to manage the values for the metadata attributes you want to associate with the object group. The Metadata Schema page enables you to manage metadata attributes.

1. Click in the Actions column to display the Edit Metadata pop-up.
2. Click to make the Values field editable.
3. In the **Values** field, enter the value for the attribute with the applicable attribute type specific format.

- **Date/Time**: Must have a valid date and time format and can use natural language processing such as; 3 Thursdays ago at 5pm.
- **Integer**: Type: Value must be numeric.
- **IP Address**: Must use one of the following formats.
  - **IPv4**: for example, **10.1.1.100** or 172.16.254.1
  - **IPv6**: supports Zero Suppression format. For example, 2001:db8::1234::567:8:1 or 2601::0800:200c:417a
- **Latitude and Longitude**: Must have valid coordinates that are decimal values: -90.00 to 90.00 values for Latitude and -180.00 to 180.00 for Longitude
- **MAC Address**: Must use the following format: 0A:00:27:00:00:00
- **Text (Validated)**: Supports up to 1024 UTF-16 characters including PCRE regex that uses preg-match (perform a regular expression match) to validate the regular expression you enter against the attribute definition from the Metadata Schema page.
- **Text**: Supports up to 65K UTF-16 varchar characters.
- **URL**: Complete the following fields:
  - **Link Display Text:** Enter the text to display in reports as the link caption.
  - **URL:** Enter the URL. Must have FQDN validation, supports username prefix, ports, protocol AND ?/& for HTTP GET variables, and optional additional PCRE regex for validation, and must be fewer than 255 characters.

4. Click **Update** to save the value.

13.2 Object Group Membership Rules

Click on an object group in the hierarchy to display the object group membership rules and the list of objects that are members of the group on the right. Membership rules enable you to automatically add objects to the object group. Rules are case sensitive and you can use Perl regular expressions to create rules.

1. Click **Add Rule** or  to display the Add/Edit Object Group Membership Rule pop-up.
2. Click the **Device Group** drop-down and select a device group/device type.
3. In the **Object Name** field, enter the object name trigger.

   Example: Enter Gig to add objects with Gig in the name to the object group.

4. In the **Description** field, enter the object description trigger.

   Example: Enter WAN to add all objects with WAN in the description to the object group.

5. Click the **Plugin** drop-down and select a plugin.
6. Click the **Object Type** drop-down and select an object type.
7. Click the **Object Subtype** drop-down and select an object subtype.
8. Click the **Metadata Namespace** drop-down to choose the metadata namespace and create rules with Metadata information. Once the namespace is selected, **Metadata Attribute** field becomes available. Choose the metadata attribute from the drop-down. Upon selecting the Metadata Attribute, **Metadata Value** field becomes available. Enter a value in the Metadata Value field.
9. Click **Save** to save the rule criteria.
10. Repeat these steps to add additional rules to use the OR Boolean.
11. Select the check box for each rule to copy, click  , and select **Copy Rules** to make a copy of the rules you select.
12. Click **Apply Rules** to add the objects that meet the rule criteria to the object group.

    The standard quantifiers in regular expressions (regex) are greedy, meaning they match as much as they can, only giving back as necessary to match the remainder of the regex. By using a lazy quantifier, the expression tries the minimal match first. You are advised to use lazy quantifier.

13.3 Object Group Membership

The Membership section displays the objects that are members of the object group. When you pin an object to an object group, rules do not affect the object's membership and you must manually unpin the object to remove its membership. When a rule adds an object to an object group, if you change the rule, the objects that were added by the rule can automatically be removed from the object group.

-  - Indicates the object is a member that was pinned to the object group.

-  – Indicates the object is a member that was added by an object group membership rule.

13.3.1 Membership Management

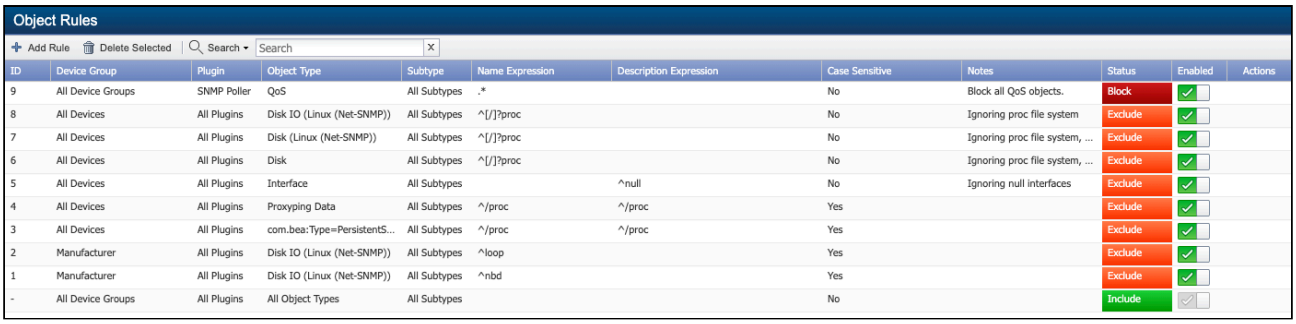
The **Pin Objects** button enables you to pin objects to the object group and  enables you to pin or unpin objects that are members of the group.

- Click **Pin Objects** to display the Pin Objects pop-up.
    a. Click the **Select Device** drop-down and select a device to display its objects.
    b. Select the check box for each object to pin to the object group.
    c. Click **Pin to Group** to pin the objects you select to the object group.
- In the Membership list, select the check box for each object to pin to the object group, click  , and select **Pin Objects** to pin the objects you select to the object group.
- Select the check box for each object to unpin from the object group, click  , and select **Unpin Objects** to unpin the objects you select from the object group. If you pin an object that was added by a rule, when you unpin the object it is removed from the membership list. If you click Apply Rule the object appears in the list again.
- In the **Device Name** column, click on a device name to display a link to the Device Summary and links to applicable report templates. For details, please refer to section **Device Summary** in *SevOne NMS User Guide*.
- In the **Object Name** column, click on an object name to display a link to the Object Summary and links to applicable report templates. For details, please refer to section **Object Summary** in *SevOne NMS User Guide*.

14 Object Rules

The Object Rules page enables you to define rules to manage the polling of objects. SevOne NMS can monitor virtually everything in your network with minimal user input. The data from some objects may not be useful for you.

To access the Object Rule page from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **Object Rules**.



ID	Device Group	Plugin	Object Type	Subtype	Name Expression	Description Expression	Case Sensitive	Notes	Status	Enabled	Actions
9	All Device Groups	SNMP Poller	QoS	All Subtypes	.*		No	Block all QoS objects.	Block		
8	All Devices	All Plugins	Disk IO (Linux (Net-SNMP))	All Subtypes	^[/]?proc		No	Ignoring proc file system	Exclude		
7	All Devices	All Plugins	Disk (Linux (Net-SNMP))	All Subtypes	^[/]?proc		No	Ignoring proc file system, ...	Exclude		
6	All Devices	All Plugins	Disk	All Subtypes	^[/]?proc		No	Ignoring proc file system, ...	Exclude		
5	All Devices	All Plugins	Interface	All Subtypes		^null	No	Ignoring null interfaces	Exclude		
4	All Devices	All Plugins	Proxyping Data	All Subtypes	^/proc	^/proc	Yes		Exclude		
3	All Devices	All Plugins	com.bea:Type=PersistentS...	All Subtypes	^/proc	^/proc	Yes		Exclude		
2	Manufacturer	All Plugins	Disk IO (Linux (Net-SNMP))	All Subtypes	^loop		Yes		Exclude		
1	Manufacturer	All Plugins	Disk IO (Linux (Net-SNMP))	All Subtypes	^nbd		Yes		Exclude		
-	All Device Groups	All Plugins	All Object Types	All Subtypes			No		Include		

14.1 Object Rules List

Rules are applied in the sequence in which they appear in the list from top to bottom. The rule list stops (i.e. short-circuits) when a rule is matched.

Click and in the Actions column to change the rule sequence. The list displays the following information.

- **Device Group** - Displays the device group/device type to which the rule applies.
- **Plugin** - Displays the name of the plugin to which the rule applies.
- **Object Type** - Displays the name of the object type to which the rule applies and displays the rule conditions.
- **Subtype** - Displays the name of the object subtype to which the rule applies.
- **Name Expression** - Displays the Perl Regular Expressions applied to the object name used to define the rule.
- **Description Expression** - Displays the Perl Regular Expressions applied to the object type description used to define the rule.
- **Case Sensitive** - Displays *Yes* if the rule is case sensitive or displays *No* if the rule is not case sensitive.
- **Notes** - Displays notes you enter for the rule to explain the purpose of the rule.
- **Status** - Displays whether discovery should include, exclude, or block objects to which the rule applies.

  - **Exclude** means that the object will be disabled by the rule. However, the object will be stored in the database.
  - **Include** means that the object will be included by the rule.
  - **Block** means that the object goes through the discovery process however, no information of the object discovered will be stored in the database.

  Important considerations

  - The advantage of **Exclude** over **Block** is that it allows you to enable the object without re-discovery. For example, for troubleshooting purposes.
  - The disadvantage of **Exclude** over **Block** is that it requires (minimal) additional storage for objects. As a rule of thumb, you can have approximately 3x your object license limit of disabled objects. For example, PAS200k can have up to 600k disabled objects.

- **Enabled** - Displays the status of the rule. Rules can be enabled or disabled at your discretion.

14.2 Manage Object Rules

You create object rules to define exceptions to the normal object discovery. The global rule to include the All Device Groups, appears at the bottom of the list and you cannot edit, delete, or move the global rule.

Examples

- The /proc file system on the Linux operating system is a read-only system-level file system that is always full, and it is rarely important to monitor. There are policies set up to alert on full file systems and you would always receive an alert for the /proc file system. You would have to disable the thresholds to prevent this. A better option is to define an object rule to block or exclude polling these objects.
- It makes sense to disable unused interfaces that are administratively up but not actually in use, such as Un-routed *VLAN* interfaces.

The general rule denoted by a '**-**' cannot be edited or deleted.

The rules you define on the Object Rules page override the interface synchronization settings you define on the Cluster Manager > **Cluster Settings** tab and in the SNMP plugin definition on the Edit Device page. For details, please refer to section **Edit Device** in *SevOne NMS User Guide*.

If you create a rule to "block all /proc" file systems, and then you delete the "block all /procs" rule, all "/proc" file system objects remain blocked until the next discovery.

1. Click **Add Rule** or click  to display the Add/Edit Rule pop-up.

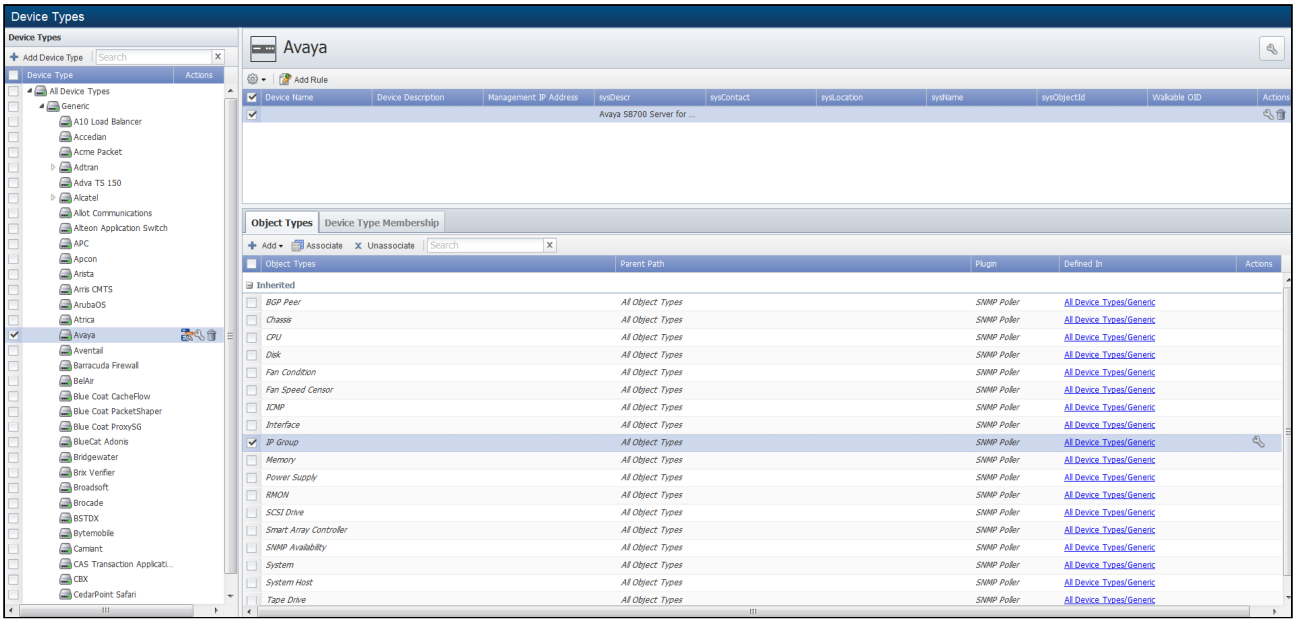
   Object rules can **exclude**, **include**, or **block** the polling of objects.

2. Click the **Device Group or Type** drop-down and select the device group/device type to which to apply the rule.
3. Click the **Plugin** drop-down and select the plugin that polls the object type to report on.
4. Click the **Object Type** drop-down and select an object type.
5. Click the **Subtype** drop-down and select an object subtype, when applicable.
6. Click the **Match the object name with this expression** drop-down
   - Select **Match** to define the rule to apply when the object name expression matches the expression you enter in the text field.
   - Select **Do Not Match** to define the rule to apply when the object name expression does not match the expression you enter in the text field.
7. In the text field enter the Perl Regular Expressions to either match or not match. Leave this field blank to not use the object name expression in the rule.
8. Click the **Match the object description with this expression** drop-down.
   - Select **Match** to define the rule to apply when the object description expression matches the expression you enter in the text field.
   - Select **Do Not Match** to define the rule to apply when the object description expression does not match the expression you enter in the text field.
9. In the text field enter the Perl Regular Expressions to either match or not match. Leave this field blank to not use the object name expression in the rule.
10. Select the **Case Sensitive** check box to apply the rule only when the Perl regular expression matches/does not match including upper and lower case of the letters you enter.
11. Click the **Enabled** check box to enable the rule.
12. Click the **Status** drop-down and select **Include**, **Exclude**, or **Block**.
13. In the **Notes** field, enter a note to associate with the rule.
14. Click **Save**.

15 Device Types

The Device Types page enables you to use discovery to classify and organize devices. Starter set device types use topology metadata driven rules to automatically add devices that topology sources discover to applicable device types. Device types enable you to associate a collection of SNMP object types to multiple devices. Each device can belong to multiple device types. Device types enable you to organize devices for SNMP polling purposes which expedites policy definition and enables you to run manufacturer independent reports for similar but not identical objects. You can associate an icon to each device type for topology reports. Device types appear in all Device Group lists and provide an additional method to secure, sort, and filter devices. For details, please refer to section **Device Group** in *SevOne NMS User Guide*.

To access the Device Types page from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **Device Types**.



15.1 Device Type Hierarchy

The device type hierarchy appears on the left. First level device types display in alphabetical order under the parent All Device Types.

1. Select the device type under which to add a new device type.
2. Click **Add Device Types** to display a pop-up.
    a. In the **Device Type Name** field, enter the device type name.
    b. Click **Save** on the pop-up.
3. - Click to edit the values for the metadata attributes you want to associate with the device types. See the Manage Metadata Values section below.
4. - Click to edit a device type name.
5. Drag and drop device types to different places in the custom device type hierarchy.
6. - Click to delete the device group. When you delete a device group that has associated policies, a message appears to inform you that associated policies will be deleted.
    a. On the message, click **View Policies** to display the Policy Association pop-up that lists the policies associated to the device group.
    b. Click **Done** to move the policies.
7. Click above the right side of the page to display the Upload Device Type Icon pop-up that enables you to associate an icon to the device type for display in Topology reports - please see section **Report Interactions** in *SevOne NMS User Guide*. Please refer to section Device Type Icons below for the list of starter set icons.
    a. Click the **Select File** to display the File Upload pop-up.
    b. Navigate your file hierarchy and select the image file to upload.
    c. Click **Open** to return to the Upload Device Type Icon pop-up.
    d. Click **Save** to associate the image with the device type or click **Remove Icon** to remove the icon associated with the device type.

15.1.1 Manage Metadata Values

The  in the device type hierarchy Actions column provides access to the Edit Metadata pop-up that enables you to manage the values for the metadata attributes you want to associate with the device type. The Metadata Schema page enables you to manage metadata attributes.

1. Click  in the Actions column to display the Edit Metadata pop-up.
2. Click  to make the Values field editable.
3. In the **Values** field, enter the value for the attribute with the applicable attribute type specific format.
    - **Date/Time**: Must have a valid date and time format and can use natural language processing such as; 3 Thursdays ago at 5pm.
    - **Integer**: Type: Value must be numeric.
    - **IP Address**: Must use one of the following formats:
        - **IPv4**: for example, **10.1.1.100** or 172.16.254.1
        - **IPv6**: supports Zero Suppression format. For example, **2001:db8::1234::567:8:1** or **2601::0800:200c:417a**
    - **Latitude and Longitude**: Must have valid coordinates that are decimal values: -90.00 to 90.00 values for Latitude and -180.00 to 180.00 for Longitude
    - **MAC Address**: Must use the following format: 0A:00:27:00:00:00
    - **Text (Validated)**: Supports up to 1024 UTF-16 characters including PCRE regex that uses preg-match (perform a regular expression match) to validate the regular expression you enter against the attribute definition from the Metadata Schema page.
    - **Text**: Supports up to 65K UTF-16 varchar characters.
    - **URL**: Complete the following fields:
        - **Link Display Text:** Enter the text to display in reports as the link caption.
        - **URL:** Enter the URL. Must have FQDN validation, supports username prefix, ports, protocol AND ?/& for HTTP GET variables, and optional additional PCRE regex for validation, and must be fewer than 255 characters.
4. Click **Update** to save the value.

15.2 Device Type Membership Rules

The Device Type Membership Rules section enables you to define rules to automatically add devices to a device type from across multiple manufacturers/products that implement the *same* MIBs. Starter set rules automatically add devices to applicable device types for Topology reports - for details, please see section **Report Interactions** in *SevOne NMS User Guide*. Device type rules run during discovery to add the devices from which the SNMP plugin can poll the metrics you specify to the device types you create. You can manually pin devices to device types and the Device Manager also enables you to pin devices to device types. Rules are case sensitive and you can use Perl regular expressions to create rules.

1. Click on a device type in the hierarchy to display the device type membership rules in the upper section on the right.
2. Click **Add Rule** or click  to display the Add/Edit Device Type Membership Rule pop-up. Each rule requires only one field. If you populate several fields per rule, the criteria you enter in that rule are cumulative.

> **Example:** Enter *Remote* in the Name field and enter *^192\.168* in the Management IP Address field in the same rule criteria row. The device must have both Remote in the name AND an IP address that starts with 192.168 to be added to the device type.

    - **Device Name** - The device name (wildcards are implied).
    - **Device Description** - The device description.
    - **Management IP Address** - The IP address of the device (^192\.168 adds all devices whose IP address starts with 192.168).
    - **sysDescr** - The text you get when you SNMP walk the sysDescr OID.
    - **sysContact** - The text you get when you SNMP walk the sysContact OID.
    - **sysLocation** - The text you get when you SNMP walk the sysLocation OID.
    - **sysName** - The text you get when you walk the sysName OID.
    - **sysObjectID** - The text you get when you SNMP walk the sysObjectID OID.
    - **Walkable OID** - Enter the number of the OID to walk. Make sure that the OID can actually be walked. You may be able to *get* an OID and get a response. However, if you *walk* an OID and the result from the **snmpgetnext** falls outside of the OID tree, discovery won't add the device to the device type.

> **Example**
>
> getnext .1.3.6.1.2.1.1.1.0 – get .1.3.6.1.2.1.1.2.0 - does not match the pattern .1.3.6.1.2.1.1.1.0.X

getnext .1.3.6.1.2.1.1.1 – get .1.3.6.1.2.1.1.1.1 - matches the pattern .1.3.6.1.2.1.1.1.X

3. Click **Save** to save the rule criteria.
4. Repeat these steps to add additional rules to use the OR Boolean.

Example: Enter *Remote* in the Name field in the first row in the rule table then click Create Rule. Enter *^192\.168* in the second row in the rule table. The device can have either Remote in the name OR an IP address that starts with 192.168 to be added to the device type.

15.3 Object Types

The Object Types tab in the lower right section contains two groups of object types Inherited and Local. The Object Types tab displays the following information.

- **Object Types** - Displays the list of object types that are associated to the device type.
- **Parent Path** - Displays where the object type resides within the object type hierarchy.
- **Plugin** - Displays the name of the plugin that polls the object type.
- **Defined In** - Displays *Local* for object types that are defined at this level in the Device Type hierarchy or displays a link that navigates you to the level in the Device Types hierarchy from where the object type inherits its definition.

The Object Types tab enables you to associate the SNMP object types you want the SNMP plugin to attempt to poll on the devices that are members of the device type.

1. On the Object Types tab, click **Associate** to display the Associate pop-up.
2. Select the check box for each object type to poll on the devices that are members of the device type.
3. Click **Associate** on the pop-up to associate the object types to the device type.

15.3.1 Manage SNMP Object Types

The Object Types tab duplicates the functionality of Object Types page to enable you to add and edit SNMP object types. For additional details see the .SNMP v5.6 topic. For interfaces and most SNMP objects, there is an OID that is dedicated to provide the index number.

Example: Walk ifIndex to yield the following

RFC1213-MIB::ifIndex.1 = INTEGER: 1
RFC1213-MIB::ifIndex.2 = INTEGER: 2
RFC1213-MIB::ifIndex.8 = INTEGER: 8

Each field on the Add/Edit SNMP Object Type pop-up has a corresponding check box on the right side to enable you to make changes at this level of the hierarchy and below. The changes you make when you select the right-hand check box override and do not affect the parent object type definition.

1. Click **Add** then **SNMP** or click  to display the Add/Edit SNMP Object Type pop-up.
2. In the **Name** field, enter the object type name.
3. Click the **Indexed By**  to display the SNMP OID Browser where you select the index OID.
4. Select the **Reverse Engineer** check box to have instances of this object type be uniquely identified by evaluating the OID of the SNMP object specified in the Indexed By field, as opposed to its value. How the values encoded within the OID are evaluated is based on the configuration of the Index Keys field. You should leave this check box selected for the vast majority of object types.
5. The **Index Keys** fields enable you to select the index keys to use to determine how to treat the remaining octets after the index. In the Possible Values field, select index keys to assign to the object type (use Ctrl or Shift keys to multi-select) then move the index keys to the Index Keys field. Index keys in the Index Keys field are assigned to the object type and they display in the sequence in which they appear listed. Possible values include the following:
    - **Integer** - A single number that indicates there is a constant amount of numbers following each OID.
    - **String** - A string prefixed with the string length. This typically appears with double quotes.
    - **String (Implied)** - A string with no length information. This must only occur as the last index value.
    - **Variable** - A variable amount of numbers prefixed with the amount of items. This is typically used for IPv4 versus IPv6 indexes.
    - **Variable (Implied)** - A Variable amount of numbers, but with no length information. This must occur as the last index value. This can be used to *eat up* the remainder of the index.

6. Click the **Name Expression** to display the SNMP OID Browser where you select the OID that results in a unique name for all object types on a device.
7. Click the **Description Expression** to display the SNMP OID Browser where you select the OID to add additional information about the object type.
8. Click the **Subtype** to display the SNMP OID Browser where you select the OID to define a subtype for the object type (used for thresholds and reports). This can generate the following variables:
    - - [TYPE]: The numerical value of the subtype.
    - - [TYPE]: The name of the subtype.
    - - [TYPE]: The description of the subtype.
9. Click the **Assert** to display the SNMP OID Browser where you select the OID to use in the assert expression that generates a list of individual object indexes. This is skipped if the object does not pass the assert expression. No variables are generated.
10. Click the **Last-change OID** to display the SNMP OID Browser where you select the OID to use to determine if a change was made to the object type since it was last polled. If the object type changed, the SNMP plugin invalidates the current data.
11. Click the **Admin-status** to display the SNMP OID Browser where you select the OID to use to determine the administrative status of the object.
12. Click the **Oper-status** to display the SNMP OID Browser where you select the OID to use to determine the operational status of the object.
13. In the **Variable** field, enter the variables, expressions, and operators you want to use to evaluate first for use with the other fields.
14. Click **Edit Subtypes** to display the Object Subtype Manager where you manage the object subtypes.
15. Click **Save**.

15.4 Device Type Membership

The Device Type Membership tab displays the devices that are members of the device type. When you pin a device to a device type, rules do not affect the device's membership and you must manually unpin the device to remove its membership. When a rule adds a device to a device type, if you change the rule, the devices that were added by the rule can automatically be removed from the device type.

-  - Indicates the device is a member that was pinned to the device type at this level.
-  - Indicates the device is a member that was added by a device type membership rule at this level.

15.4.1 Membership Management

The following tools enable you to manage the devices that are members of the device type you select in the hierarchy.

- Click **Pin Devices** to display a pop-up that displays all devices in SevOne NMS.
    a. Select the check box for each device to pin to the device type.
    b. Click **Pin to Type** on the pop-up to pin the devices you select to the device type.
- Select the check box for each device to unpin from the device type and click **Unpin Devices**. If you pin a device that was added by a rule, when you unpin the device it is removed from the membership list. If you do not change the rule, the device appears in the list again upon the next discovery.
- Click  in the Actions column to display the Edit Metadata pop-up. See the Manage Metadata Values section above.

15.5 Device Type Icons

Topology related device types use the following icons by default. You can change the icon from the  as described above.

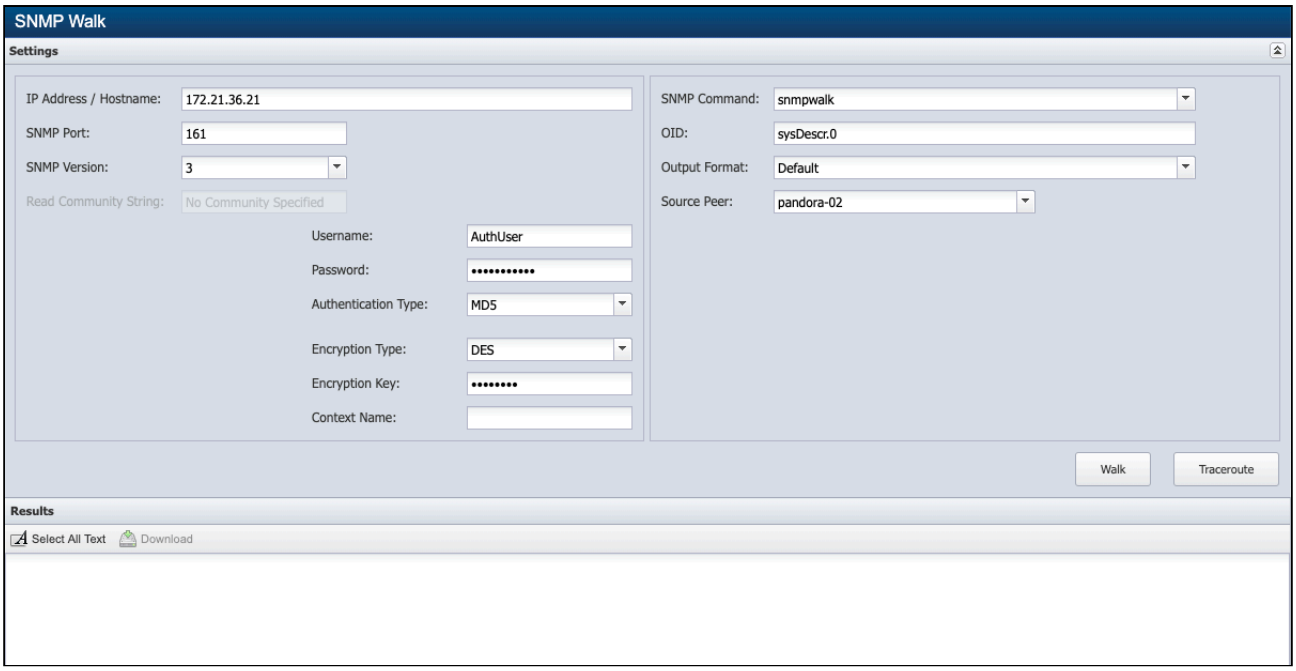
 - Bridge	 - Cloud	 - Cloud No Access	 - Computer

 - Computers	 - Controller	 - Device	 - Firewall
 - Host	 - PBX	 - Printer	 - Rack
 - Switch	 - UPS	 - User	 - Users
 - VM	 - Wireless		

16 SNMP Walk

The SNMP Walk page enables you to discover and certify SNMP MIBs and to troubleshoot SNMP connectivity problems. For details, please refer to the SNMP topic.

To access the SNMP Walk page from the navigation bar, click the **Devices** menu and select **SNMP Walk**. The Device Manager also provides access to the SNMP Walk.



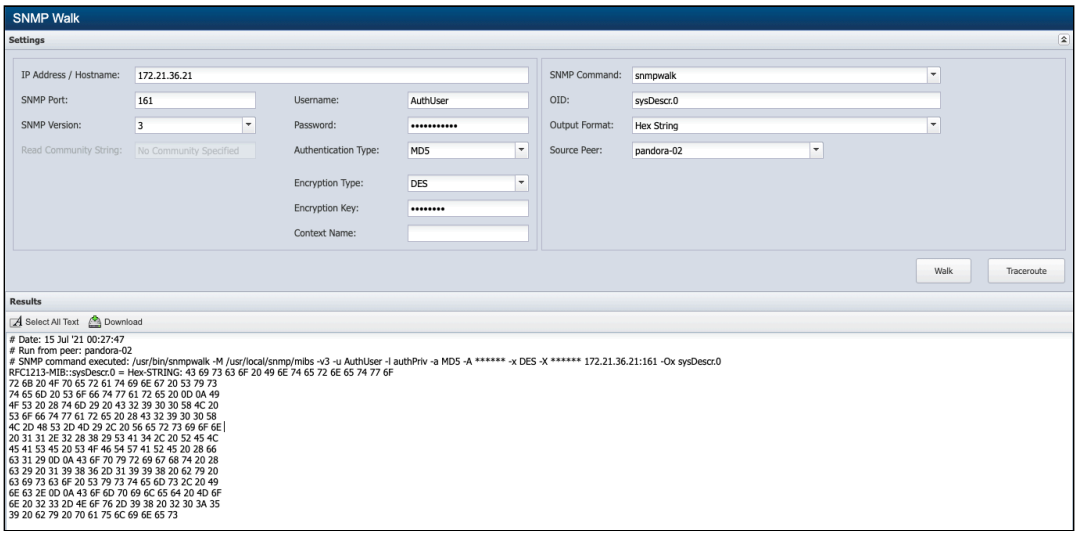
16.1 Perform an SNMP Walk / Traceroute

When you access the SNMP Walk page from the Device Manager, some of the following fields are pre-populated.

1. In the **IP Address/Hostname** field, enter the IP address or Hostname of the device.
2. In the **SNMP Port** field, enter the port on which the device is listening for SNMP traffic.
3. Click the **SNMP Version** drop-down and select the SNMP version of the device.
    - If you select SNMP Version 1 or 2c, in the **Read Community String** field, enter the read only community string SevOne NMS needs to authenticate onto the device if the string is different from what you enter on the Cluster Manager. Leave clear to use the string from the Cluster Manager > **Cluster Settings** tab.
    - If you select Version 3, complete the following fields.
        i. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the device.
        ii. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the device.
        iii. Click the **Authentication Type** drop-down.
            - Select **None** (usmNoAuthProtocol) to not use an authentication method to send or receive messages.
            - Select **MD5** (usmHMACMD5AuthProtocol) to use MD5 authentication protocol for messages.
            - Select **SHA** (usmHMACSHAAuthProtocol) to use SHA authentication protocol for messages.
        iv. If you select MD5 or SHA in the previous step, click the **Encryption Type** drop-down.
            - Select **None** to not use encryption to send or receive messages.
            - Select **AES** to use the Advanced Encryption Standard encryption method.
            - Select **DES** to use the Data Encryption Standard encryption method.
        v. If you select AES or DES in the previous step, in the **Encryption Key** field, enter the localized key the authentication protocol on the device requires to authenticate messages.
        vi. In the **Context Name** field, enter the context name. Several vendors implement SNMP with Context to access certain objects, typically to access virtual objects. Vendor may choose checkpoint firewalls to access SNMP stats from the virtual firewalls.
4. Click the **SNMP Command** drop-down.
    - Select **snmpwalk** to query a device for an OID and all of its conceptual children.
    - Select **snmpget** to query the device for an OID.

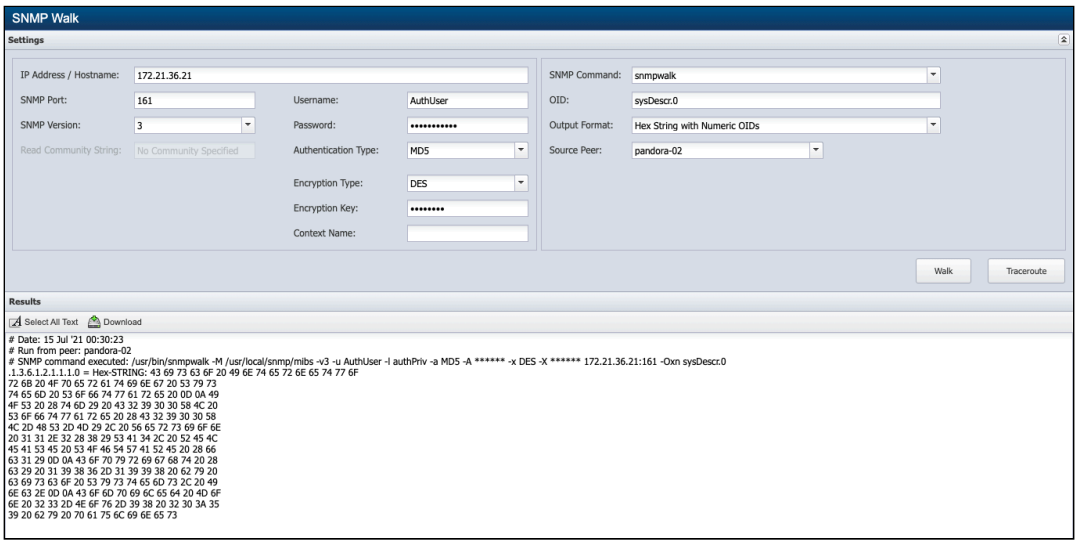
5. In the **OID** field, enter the OID to walk. If you leave this to the default OID, the walk may take a while. Be specific with your search. You should use ".1.3" or ".1.3.6" to avoid partial or broken walks.
6. Click the **Output Format** drop-down.
    - Select **Default** to display OIDs in text format (e.g. IF-MIB::ifInErrors.1).
    - Select **Default With Numeric Indexes** to translate strings to ASCII numeric values.
    - Select **Numeric OIDs** to translate OIDs into numeric format (e.g. .1.3.6.1.2.1.2.2.1.14.2).
    - Select **Certification Walk** to translate OIDs into a form that SevOne Support Engineers can use to perform device certifications.
    - Select **Hex String** to display OIDs in text format and the output in HEX format.

Example



- Select **Hex String with Numeric OIDs** to translate OIDs into numeric format and the output in HEX format.

Example



7. Click the **Source Peer** drop-down and select the peer to perform the walk.
8. Click **Walk** to perform the SNMP walk or **Traceroute** to allow troubleshooting directly from within the appliance.

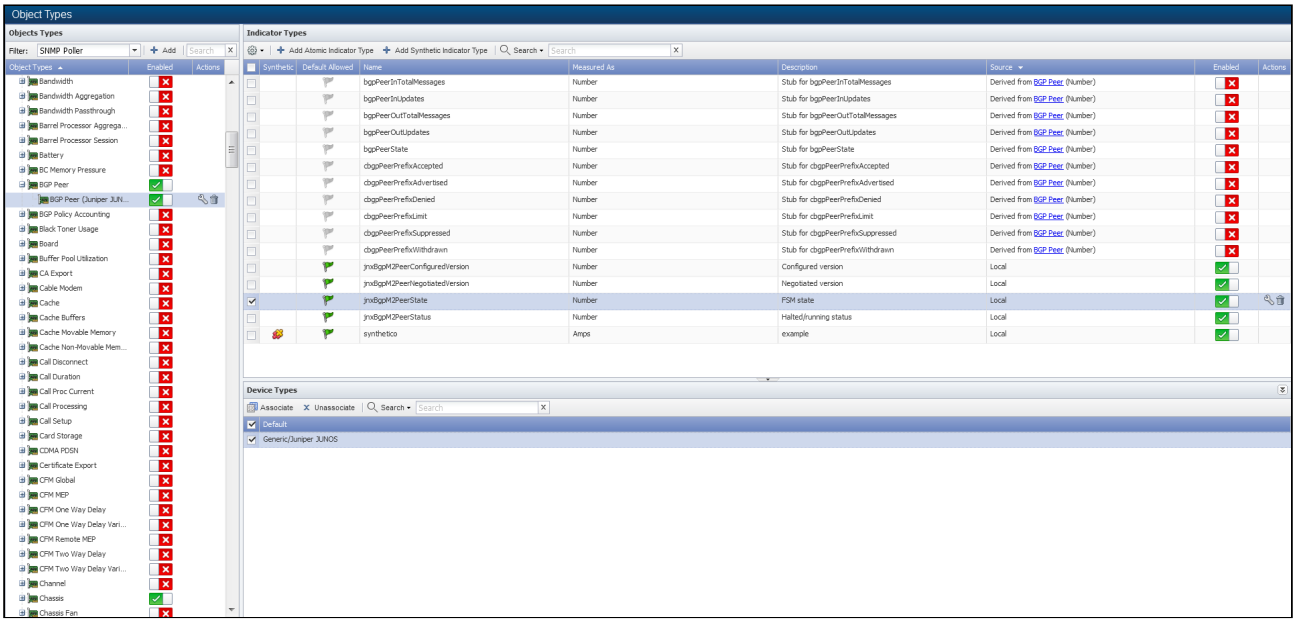
> SNMP Walk / Traceroute on multiple devices in one instance is **not** allowed.

9. After walk or traceroute is performed, **Download** button under **Results** is available.
10. Click **Select All Text** to enable you to copy the results to your computer clipboard so you can paste the results into an email or document etc.
11. Click **Download** to convert the walk / traceroute results to a .txt file you can save to your local machine.

17 Object Types

The Object Types page enables you to manage the object types and the indicator types you want each plugin to poll. Object types enable a plugin to discover related objects on a device. Indicator types enable a plugin to collect data from the indicators on objects. SevOne NMS provides a starter set of object types and indicator types. SevOne NMS discovers devices and discovery manages the objects on the devices based on the plugins that are enabled for each device. Objects contain indicators that are polled to gather the measurement points of the physical and logical capabilities of a device.

To access the Object Types page from the navigation bar, click the **Administration** menu and select **Monitoring Configuration**, then **Object Types**.



In **Object Types** and **Indicator Types**, you can populate expressions in various fields. When variable(s) are evaluated using the S3 sytax, the content of the variable must not be a string OID. Rather, OID must be in numerical form. During Discovery, untranslated string OIDs do not return any results. However, OID literals in the string form are accepted. For details on S3 syntax, please refer to *SNMP Quick Start Guide*.

17.1 Object Types

Object types determine what objects are discovered on a device. Object discovery varies from plugin to plugin. You must enable object types for the JMX plugin and the WMI plugin on the Object Types page to discover JMX and WMI objects. You define objects for the Calculation plugin on the Calculation Editor. You define objects for the xStats plugin on the xStats Source Manager.

The Object Rules page enables you to define rules to disable, enable, or block discovery of objects and the Object Manager enables you to manage the objects on each device. The Cluster Manager > **Cluster** Settings tab enables you to disable polling of objects the SNMP plugin discovers to be operationally down or administratively down.

The Object Types list appears on the left and displays the object types for the filter you select above the list. Disabled object types are not discovered or polled and disabled objects do not count towards your license capacity. See the following sections for plugin specific object type management steps.

- **Filter** - Click the **Filter** drop-down and select the plugin whose object types you want to manage. Please see the plugin specific sections below.
- **Add** and  - Click to display the plugin specific object type pop-up that enables you to add or edit the object type. Please see the plugin specific sections below.
- **Object Types** - Displays the list of object type names. Click an object type name to display the associated indicator types and device types on the right. Names with ***bold italic*** font cannot be edited or deleted.
- **Enabled** - Displays one of the following:
  -  - Object type is enabled and the objects in the object type can be polled on devices for which you enable the plugin.

-  - Object type is disabled and no metrics are collected.
-  - Object type is enabled and required for the plugin. You cannot disable, edit, or delete the object type. This object type and its indicator types are polled when you enable the plugin for a device.

17.2 Indicator Types

Indicator types determine what indicators are polled for an object type. The indicator types list displays the indicator types for the object type you select in the Object Types list. Please see the plugin specific sections below. There are two types of indicator types.

- Atomic indicator types are measured directly by the plugin.
- Synthetic indicator types are indicators whose value is dependent upon other indicators; atomic and/or synthetic. Synthetic indicators enable you to combine the data from several indicators into one synthetic indicator so that SevOne NMS can properly evaluate indicators such as Percentage Loss, Percent Error, Percent Idle, and other high precision metrics.

The Indicator Types section provides controls to manage indicator types and displays the following information.

-  - Select the check box for each indicator type to manage, click  , and select one of the following options.
  - Select **Delete** to delete the indicator types and all associated indicators.
  - Select **Turn On for Every Device** to allow all devices, for which you enable the applicable plugin, to discover the indicators associated with the indicator types at the next discovery and poll data. You can only turn on indicator types that display  in the Enabled column.
  - Select **Turn Off for Every Device** to stop polling the indicators associated with the indicator types on all devices at the next discovery.
  - Select **Enable** to enable the ability to poll all indicators associated with the indicator types.
  - Select **Disable** to prevent the ability to poll the indicators associated with the indicator types.
  - Select **Convert to Atomic** to convert a synthetic indicator type to an atomic indicator type.
  - Select **Convert to Synthetic** to convert an atomic indicator type to a synthetic indicator type.
  - Select **Implement** to maintain the indicator types of an SNMP poller child level object type independently from the indicator type whose definition it derived from a higher level in the SNMP object type hierarchy.
- **Add Atomic Indicator** and  - Atomic indicators are measured directly by the plugin. Please see the plugin specific sections below.
- **Add Synthetic Indicator Type** and  - Synthetic indicators enable you to perform math on multiple metrics collected from multiple indicators on a single monitored object in order to calculate new KPIs. You can define synthetic indicator types for the following plugins: Calculation, Deferred Data, JMX, SNMP, WMI, and xStats. Please refer the Synthetic Indicator Types section.
- **Synthetic** - Displays  if the indicator type is synthetic. The column appears clear if the indicator type is atomic. Please see the Synthetic Indicator Types section.
- **Default Allowed** - Indicates if the plugin polls the indicator type by default on devices when you enable the plugin for a device. Click the icon to change the Default Allowed setting.
  -  - The plugin attempts to poll the indicator type by default on devices when you enable the plugin for a device. The indicator type appears with a check mark on the Indicator Type Maps page.
  - - You must manually enable the plugin to poll the indicator type for each device from the Edit Device page on the Indicator Type Maps page.
  -  - Indicator type is disabled and the plugin does not poll the indicator type. If you enable the indicator type, the plugin attempts to poll the indicator type by default on devices when you enable the plugin for a device.
- **Name** - Displays the indicator type name.
- **Description** - Displays the indicator type description.
- **Source** - Displays *Local* for indicator types that have not inherited their definition from a higher level object type. Displays *Implemented From <name>* for indicator types that have inherited aspects from an object type that is higher up in the Object Types hierarchy. The Name of the implementing object type appears as a link that navigates you to the object type from which the indicator type derived its definition.
- **Enabled** - Displays one of the following:
  -  - Indicator type is enabled and, if the object type is enabled, the indicator type can be polled on devices for which you enable the plugin.
  -  - Indicator type is disabled and no data is collected.
  -  - Indicator type is enabled and required for the plugin. You cannot disable, edit, or delete the indicator type. When you enable the plugin for a device, the indicators for this indicator type are polled.

17.3 Manage Object Types and Atomic Indicator Types

Object types and indicator types are plugin specific. You can add, edit, and delete object types and indicator types for the following plugins: Calculation, Deferred Data, JMX, SNMP, WMI, and xStats. You can view details for the object types and indicator types for the other plugins to determine if the data they poll is relevant to your network monitoring requirements. Please see the following plugin specific sections for object type management steps and for atomic indicator type management steps. Please refer to Synthetic Indicator Types section for synthetic indicator type management steps.

17.3.1 Calculation Poller

The Calculation plugin polls the indicator types for the object types you enable. Calculation objects are composed of variables that represent the calculations of metrics other plugins poll from multiple indicators. To poll Calculation data, define the Calculation object types on this page, define Calculation objects on the Calculation Editor page, and enable the Calculation plugin for the device on the New Device page or on the Edit Device page.

17.3.1.1 Calculation Object Types

Perform the following steps to manage Calculation object types. Discovery uses the object "objectTypeId" && "group_id" to determine if a Calculation object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **Calculation Poller** to display the Calculation object types in the Object Types list.
2. Click **Add** or click  to display the Add/Edit Calculation Object Type pop-up.
3. In the **Name** field, enter the Calculation object type name.
4. Select the check box for the **Note** field to enable it for editing. Enter any additional information you would like to include.
5. Click **Save**.

17.3.1.2 Calculation Atomic Indicator Types

Perform the following steps to manage Calculation atomic indicator types. For Calculation synthetic indicator types, please refer to Synthetic Indicator Types section below.

1. Click the **Filter** drop-down and select **Calculation Poller** to display the Calculation object types in the Object Types hierarchy.
2. Click on an object type to display its indicator types on the right.
3. Click **Add Atomic Indicator Type** or click  next to an atomic indicator type to display the Add/Edit Calculation Indicator Type pop-up.
4. In the **Indicator Name** field, enter the name of the indicator type.
5. In the **Description** field, enter the name to display.
6. The **Indicator Type** field displays Gauge. All Calculation indicator types are gauges.
7. Click the **Measured As** drop-down and select the unit of measure for the indicator type.
8. Click the **Display As** drop-down and select the unit of measure to display for the indicator type.
9. Select the **Default allowed for new devices** check box to have the Calculation plugin poll the indicator type by default when the object type is enabled and you enable the Calculation plugin for a device.
10. In the **Note** field, enter any additional information you would like to include.
11. Click **Save**.

The Device Types section is irrelevant for Calculation poller object types.

17.3.2 Deferred Data

The Deferred Data object types are enabled by default. The Deferred Data plugin uses API scripts you define to import third party data. Contact your SevOne Sales Engineer to schedule API training. To poll deferred data devices, enable the Deferred Data plugin for the device on the New Device or on the Edit Device page.

17.3.2.1 Deferred Data Object Types

Perform the following steps to manage Deferred Data object types.

1. Click the **Filter** drop-down and select **Deferred Data** to display the Deferred Data object types in the Object Types list.
2. Click **Add** or click  to display the Add/Edit Deferred Data Object Type pop-up.
3. In the **Name** field, enter the object type name.
4. Select the check box for the **Note** field to enable it for editing. Enter any additional information you would like to include.

5. Click **Save**.

17.3.2.2 Deferred Data Atomic Indicator Types

Perform the following steps to manage Deferred Data indicator types. For Deferred Data synthetic indicator types, please see the Synthetic Indicator Types section below.

1. Click the **Filter** drop-down and select **Deferred Data** to display the Deferred Data object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click **Add Atomic Indicator Type** or click  to display the Add/Edit Deferred Data Indicator Type pop-up.
4. In the **Indicator Name** field, enter the name of the indicator type.
5. In the **Description** field, enter the name to display.
6. Click the **Indicator Type** drop-down.

- Select **Gauge** for indicators that have specific values when polled.
- Select **Counter32** for 32 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box to total the delta/differences between polls to provide the ability to graph things like the number of errors in a day, for example.
- Select **Counter64** for 64 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box.

- Click the **Measure As** drop-down and select a data unit.
- Click the **Display As** drop-down and select a display unit.
- Select the **Maximum Value** check box to indicate the indicator type has a maximum value. You must select this check box if you want to use the indicators in this indicator type for percentile metrics.
- Select the **Default allowed for new devices** check box to have the Deferred Data plugin poll the indicator type by default when the object type is enabled and you enable the Deferred Data plugin for a device.
- In the **Note** field, enter any additional information you would like to include.
- Click **Save**.

The Device Types section is irrelevant for Deferred Data poller object types.

17.3.3 DNS Object Poller

The DNS object type is enabled by default. There is one DNS object type: DNS Data. To poll DNS devices, enable the DNS plugin for the device on the New Device page or on the Edit Device page.

17.3.3.1 DNS Object Types and Indicator Types

You cannot add, edit, or delete DNS object types or DNS indicator types.

1. Click the **Filter** drop-down and select **DNS Poller** to display the DNS Data object type in the Object Types list.
2. Click on the **DNS Data** object type to display its indicator types on the right.
3. Click  to display the DNS Indicator Type pop-up.
4. View the following DNS indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Default allowed for new devices** - Check box appears selected and the DNS plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for DNS poller object types.

17.3.4 HTTP Poller

The HTTP object type is enabled by default. There is one HTTP object type: Website Data. To poll HTTP devices, enable the HTTP plugin for the device on the New Device page or on the Edit Device page. The Cluster Manager > **Peer Settings** tab enables you associate each peer with an HTTP proxy server. The Authentication Settings page enables you to upload certificates to monitor https.

17.3.4.1 HTTP Object Types and Indicator Types

You cannot add, edit, or delete HTTP object types or HTTP indicator types.

1. Click the **Filter** drop-down and select **HTTP Poller** to display the Website Data object type in the Object Types list.
2. Click on the **Website Data** object type to display its indicator types on the right.
3. Click  to display the HTTP Indicator Type pop-up.
4. View the following HTTP indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Default allowed for new devices** - Check box appears selected and the HTTP plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for HTTP poller object types.

17.3.5 ICMP Poller

The ICMP object type is enabled by default. There is one ICMP object type: Ping Data. To poll ICMP devices, enable the ICMP plugin for the device on the New Device page or on the Edit Device page.

17.3.5.1 ICMP Object Types and Indicator Types

You cannot add, edit, or delete ICMP object types or ICMP indicator types.

1. Click the **Filter** drop-down and select **ICMP Poller** to display the Ping Data object type in the Object Types list.
2. Click on the **Ping Data** object type to display its indicator types on the right.
3. Click  to display the ICMP Indicator Type pop-up.
4. View the following ICMP indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Default allowed for new devices** - Check box appears selected and the ICMP plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for ICMP poller object types.

17.3.6 IP SLA Poller

IP SLA object types are enabled by default. To poll IP SLA devices, enable the IP SLA plugin for the device on the New Device page or on the Edit Device page. The Probe Manager enables you to manage IP SLAs.

17.3.6.1 IP SLA Object Types and Indicator Types

SevOne NMS provides the following IP SLA object types.

- DHCP
- DLSw
- DNS
- Echo
- Ethernet Jitter
- Ethernet Ping
- FTP
- HTTP

- ICMP Jitter
- RTP
- TCP Connect
- UDP Echo
- UDP Jitter
- Video
- VoIP

You cannot add, edit, or delete IP SLA object types or IP SLA indicator types. Discovery uses the object "objectTypeId" && "isSevone" && ("type" && "owner" && "tag" && (! foundObject->isDuplicate || (foundObject->isDuplicate && snmpObjectId))) to determine if an IP SLA object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **IP SLA Poller** to display the IP SLA object types in the Object Types list.
2. Click on an object type display its indicator types on the right.
3. Click  to display the IP SLA Indicator Type pop-up.
4. View the following IP SLA indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Default allowed for new devices** - Check box appears selected and the IP SLA plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for IP SLA poller object types.

17.3.7 JMX Poller

JMX object types are disabled by default. The JMX plugin polls the indicator types associated with the JMX object types you enable. To poll JMX devices, enable devices to send JMX data to SevOne NMS, enable the JMX object types on this page, and enable the JMX plugin for the device on the New Device page or on the Edit Device page.

17.3.7.1 JMX Object Types

Perform the following steps to manage JMX object types. Discovery uses the object "jmx_name" to determine if a JMX object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **JMX Poller** to display the JMX object types in the Object Types list.
2. Click **Add** or click  to display the Add/Edit JMX Object Type pop-up.
3. In the **Name** field, enter the name of the object type.
4. In the **Domain** field, enter the domain of the object type.
5. In the **Type** field enter the type of the object type.
6. In the **Class Name** field, enter the class name of the object type.
7. In the **Alias** field enter the alias of the object type.
8. Select the **Enabled** check box to enable the JMX plugin to poll the indicator types in the object type.
9. Select the check box for the **Note** field to enable it for editing. Enter any additional information you would like to include.
10. Click **Save**.

17.3.7.2 JMX Atomic Indicator Types

Perform the following steps to manage JMX atomic indicator types. For JMX synthetic indicator types, please see the Synthetic Indicator Types section below.

1. Click the **Filter** drop-down and select **JMX Poller** to display the JMX object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click **Add Atomic Indicator Type** or click  to display the Add/Edit JMX Indicator Type pop-up.
4. In the **Indicator Name** field enter the name of the indicator type.
5. In the **Description** field, enter the name to display.
6. Click the **Indicator Type** drop-down.

- Select **Gauge** for indicators that have specific values when polled.

- Select **Counter32** for 32 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box to total the delta/differences between polls to provide the ability to graph things like the number of errors in a day, for example.
- Select **Counter64** for 64 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box.

- Click the **Measured As** drop-down and select a data unit.
- Click the **Display As** drop-down and select a display unit.
- Select the **Maximum Value** check box to indicate the indicator type has a maximum value. You must select this check box if you want to use the indicators in this indicator type for percentile metrics.
- Select the **Default allowed for new devices** check box to have the JMX plugin poll the indicator type by default when the object type is enabled and you enable the JMX plugin for a device.
- In the **Note** field, enter any additional information you would like to include.
- Click **Save**.

The Device Types section is irrelevant for JMX poller object types.

17.3.8 MySQL Database Poller

MySQL Database object types are enabled by default. To poll MySQL Database devices, enable the Database plugin for the device on the New Device page or on the Edit Device page.

17.3.8.1 MySQL Database Object Types and Indicator Types

SevOne NMS provides the following MySQL object types.

- MySQL Server
- My SQL Server: Command Statistics
- MySQL Server: InnoDB Statistics
- MySQL Server: SSL Statistics

You cannot add, edit, or delete MySQL Database object types or MySQL Database indicator types. Discovery uses the object "objectTypeId" && "databaseId" to determine if a MySQL object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **MySQL Database Poller** to display the MySQL Database object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click  to display the MySQL Database Indicator Type pop-up.
4. View the following MySQL Database indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to appear.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Default allowed for new devices** - Check box appears selected and the MySQL Database plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for MySQL Database poller object types.

17.3.9 NBAR Poller

The NBAR object type is enabled by default. There is one NBAR object type: NBAR Data. To poll NBAR devices, enable the device to send NBAR data to SevOne NMS and enable both the NBAR plugin and the SNMP plugin for the device on the New Device page or on the Edit Device page.

17.3.9.1 NBAR Object Types and Indicator Types

You cannot add, edit, or delete NBAR object types or NBAR indicator types. Discovery uses the object 2 of 3 match on "name", "description", and "snmpObjectId" to determine if an NBAR object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **NBAR Poller** to display the NBAR Data object type in the Object Types list.
2. Click on the **NBAR Data** object type to display its indicator types on the right.

3. Click  to display the NBAR Indicator Type pop-up.
4. View the following NBAR indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
    - **Default allowed for new devices** - Check box appears selected and the NBAR plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for NBAR poller object types.

17.3.10 Oracle Database Poller

Oracle Database object types are enabled by default. To poll Oracle Database devices, enable the Database plugin for the device on the New Device page or on the Edit Device page.

17.3.10.1 Oracle Database Object Types and Indicator Types

SevOne NMS provides the following Oracle Database object types.

- Oracle Statistics
- Oracle Tablespace

You cannot add, edit, or delete Oracle Database object types or Oracle Database indicator types. Discovery uses the object "objectTypeId" && "databaseId" && "identifier" to determine if an Oracle Database object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **Oracle Database Poller** to display the object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click  to display the Oracle Database Indicator Type pop-up.
4. View the following Oracle Database indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
    - **Default allowed for new devices** - Check box appears selected and the Oracle Database plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for Oracle Database poller object types.

17.3.11 Portshaker Poller

The Portshaker object type is enabled by default. There is one Portshaker object type: Port Data. To poll Portshaker devices, enable the Portshaker plugin for the device on the New Device page or on the Edit Device page.

17.3.11.1 Portshaker Object Types and Indicator Types

You cannot add, edit, or delete Portshaker object types or Portshaker indicator types.

1. Click the **Filter** drop-down and select **Portshaker Poller** to display the Port Data object type in the Object Types list.
2. Click on the **Port Data** object type to display its indicator types on the right.
3. Click  to display the Portshaker Indicator Type pop-up.
4. View the following Portshaker indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.

- **Description** - Displays the name to display.
- **Indicator Type** - Displays the indicator type.
- **Measure As** - Displays the indicator type data unit.
- **Display As** - Displays the indicator type display unit.
- **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
- **Default allowed for new devices** - Check box appears selected and the Portshaker plugin polls the indicator type by default.
- **Note** - Displays any additional information.

5. Click **Cancel**.

The Device Types section is irrelevant for Portshaker poller object types.

17.3.12 Process Poller

The Process object type is enabled by default. There is one Process object type: Process. To poll Process devices, enable both the Process plugin and the SNMP plugin for the device on the New Device page or on the Edit Device page. The Process plugin works in conjunction with SNMP via the `HOST-RESOURCES MIB`. The MIB stores information about all currently running processes, including their CPU time and memory utilization.

Processes group by hrSWRunName, so if 20 SSH sessions are open, they all count as one process with 20 instances. The Process plugin aggregates the values from the combined instances to present the total.

> **Example**
>
> The memory usage for each SSH session is summed together.

17.3.12.1 Process Object Types and Indicator Types

You cannot add, edit, or delete Process object types or Process indicator types.

1. Click the **Filter** drop-down and select **Process Poller** to display the Process object type in the Object Types list.
2. Click on the **Process** object type to display its indicator types on the right.
3. Click  to display the Process Indicator Type pop-up.
4. View the following Process indicator type details.
   - **Indicator Name** - Displays the name of the indicator type.
   - **Description** - Displays the name to display.
   - **Indicator Type** - Displays the indicator type.
   - **Measure As** - Displays the indicator type data unit.
   - **Display As** - Displays the indicator type display unit.
   - **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
   - **Default allowed for new devices** - Check box appears selected and the Process plugin polls the indicator type by default.
   - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for Process poller object types.

17.3.13 Proxy Ping Poller

The Proxy Ping object type is enabled by default. There is one Proxy Ping object type: Proxyping Data. To poll Proxy Ping devices, enable both the Proxy Ping plugin and the SNMP plugin for the device on the New Device page or on the Edit Device page.

17.3.13.1 Proxy Ping Object Types and Indicator Types

You cannot add, edit, or delete Proxy Ping object types or Proxy Ping indicator types.

1. Click the **Filter** drop-down and select **Proxy Ping Poller** to display the Proxyping Data object type in the Object Types list.
2. Click on the **Proxyping Data** object type to display its indicator types on the right.
3. Click  to display the Proxy Ping Indicator Type pop-up.
4. View the following Proxy Ping indicator type details.
   - **Indicator Name** - Displays the name of the indicator type.

- **Description** - Displays the name to display.
- **Indicator Type** - Displays the indicator type.
- **Measure As** - Displays the indicator type data unit.
- **Display As** - Displays the indicator type display unit.
- **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
- **Default allowed for new devices** - Check box appears selected and the Proxy Ping plugin polls the indicator type by default.
- **Note** - Displays any additional information.

5. Click **Cancel**.

The Device Types section is irrelevant for Proxy Ping poller object types.

17.3.14 SNMP Poller

SNMP object types are enabled by default. To poll SNMP devices, enable the device to send SNMP data to SevOne NMS and enable the SNMP plugin for the device on the New Device page or on the Edit Device page. The MIB Manager enables you to add MIBs which are the foundation for SNMP object types and SevOne Support Engineers can perform a device certification to enable you to add the SNMP object types that are specific to your network.

Device type associations enable you to have devices you assign to the device types you define on the Device Types page automatically attempt to poll the enabled SNMP object type enabled indicators.

The SNMP plugin uses text related to and based on OIDs to create human readable object names and descriptions. This provides the ability to use logical and mathematical expressions to store a resultant value in order to relate OIDs to each other. The SNMP plugin uses this to cross reference OIDs across the device's SNMP tree to gather descriptive information about a particular object. Please refer to SNMP topic for details.

17.3.14.1 SNMP Object Types

Perform the following steps to manage SNMP object types. For interfaces and most SNMP objects, there is an OID that is dedicated to provide the index number.

> **Example:** Walk ifIndex to yield the following:
> - RFC1213-MIB::ifIndex.1 = INTEGER: 1
> - RFC1213-MIB::ifIndex.2 = INTEGER: 2
> - RFC1213-MIB::ifIndex.8 = INTEGER: 8

SNMP object types are hierarchical. You can disable or enable object types at each level of the hierarchy which means the parent object type can be disabled while the child object type is enabled. Each field on the Add/Edit SNMP Object Type pop-up has a corresponding check box on the right side to enable you to make changes at this level of the hierarchy and below. The changes you make when you select the right-hand check box override and do not affect the parent object type definition.

1. Click the **Filter** drop-down and select **SNMP Poller** to display the SNMP object types in the Object Types hierarchy.
2. Click **Add** or click  to display the Add/Edit SNMP Object Type pop-up.

   > When adding/editing an object type at a parent level, **Indexed By** field is optional. If the Indexed By field is entered, its children will automatically inherit the OID but if needed, it can be changed. However, if the Indexed By field is not entered at the parent level then, when the child object type is created, Indexed By field for the child must be specified.

3. In the **Name** field, enter the object type name.
4. Click the **Indexed By**  to display the SNMP OID Browser where you select the index OID. Perform the following steps on the SNMP OID Browser to select an OID.
   a. In the **Search** field, enter alphanumeric text to filter the OID search.
   b. Navigate the **OID Tree** hierarchy to locate and select an OID.
   c. View the OID in the display on the right side of the SNMP OID Browser.
   d. Click **Select This OID** to select the OID.
5. Select the **Reverse Engineer** check box to have instances of this object type be uniquely identified by evaluating the OID of the SNMP object specified in the Indexed By field, as opposed to its value. How the values encoded within the OID are evaluated is based on the configuration of the Index Keys field. You should leave this check box selected for the vast majority of object types.

6. The **Index Keys** fields enable you to select the index keys to use to determine how to treat the remaining octets after the index. In the Possible Values field, select index keys to assign to the object type (use Ctrl or Shift keys to multi-select) then move the index keys to the Index Keys field. Index keys in the Index Keys field are assigned to the object type and they display in the sequence in which they appear listed. Possible values include the following:
    - **Integer** - A single number that indicates there is a constant amount of numbers following each OID.
    - **String** - A string prefixed with the string length. This typically appears with double quotes.
    - **String (Implied)** - A string with no length information. This must only occur as the last index value.
    - **Variable** - A variable amount of numbers prefixed with the amount of items. This is typically used for IPv4 versus IPv6 indexes.
    - **Variable (Implied)** - A Variable amount of numbers, but with no length information. This must occur as the last index value. This can be used to "eat up" the remainder of the index.
7. Click the **Name Expression**  to display the SNMP OID Browser where you select the OID that results in a unique name for all object types on a device.
8. Click the **Description Expression**  to display the SNMP OID Browser where you select the OID to add additional information about the object type.
9. Click the **Subtype**  to display the SNMP OID Browser where you select the OID to define a subtype for the object type (used for thresholds and reports). This can generate the following variables:
    - **-** [TYPE]: The numerical value of the subtype.
    - - [TYPE]: The name of the subtype.
    - - [TYPE]: The description of the subtype.
10. Click the **Assert**  to display the SNMP OID Browser where you select the OID to use in the assert expression that generates a list of individual object indexes. This is skipped if the object does not pass the assert expression. No variables are generated.

    > **Example:** To match only Ethernet interfaces and ignore everything else, enter the following statement.
    > `ifType == 6`
    > This separates memory from disks in the Host Resources MIB; both use hrStorageIndex, but each has a different value for hrStorageType. You should not modify the default Interface definition.

11. Click the **Last-change OID**  to display the SNMP OID Browser where you select the OID to use to determine if a change was made to the object type since it was last polled. If the object type changed, the SNMP plugin invalidates the current data.
12. Click the **Admin-status**  to display the SNMP OID Browser where you select the OID to use to determine the administrative status of the object.
13. Click the **Oper-status**  to display the SNMP OID Browser where you select the OID to use to determine the operational status of the object.
14. In the **Variable** field, enter the variables, expressions, and operators you want to use to evaluate first for use with the other fields.
15. Select the check box for the **Note** field to enable it for editing. Enter any additional information you would like to include.
16. Click **Edit Subtypes** to display the Object Subtype Manager where you manage the object subtypes.
17. Click **Save**.
18. In the Object Types hierarchy, click to select the object type to which to associate device types.
19. In the Device Type section on the right, click **Associate** to display a pop-up that enables you to associate the SNMP object type with device types on which you want to poll data.
    a. Select the check box for each device type to which to associate the object type.
    b. Click **Associate** to create the association and to close the pop-up.

17.3.14.2 SNMP Atomic Indicator Types

Perform the following steps to manage SNMP atomic indicator types. For SNMP synthetic indicator types, please see section Synthetic Indicator Types below.

1. Click the **Filter** drop-down and select **SNMP Poller** to display the SNMP object types in the Object Types hierarchy.
2. Click on an object type to display its indicator types on the right.
3. Click **Add Atomic Indicator Type** or click  next to an atomic indicator type to display the Add/Edit SNMP Indicator Type pop-up.
4. In the **Indicator Name** field, enter the name of the indicator type.
5. In the **Description** field, enter the name to display.

6. View the **Indexed By** OID you define for the object type. You cannot edit this field.
7. For certain 64 bit OIDs, click the **OID (high bit)** to display the SNMP OID Browser where you select the OID to describe the top 32 bits on the OID. Please refer to the note below.
8. Click the **OID** to display the SNMP OID Browser where you select the OID to describe the object type expression.
9. Click the **Indicator Type** drop-down.

- Select **Gauge** for indicators that have specific values when polled.
- Select **Counter32** for 32 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box to total the delta/differences between polls to provide the ability to graph things like the number of errors in a day, for example.
- Select **Counter64** for 64 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box.

- Click the **Measured As** drop-down and select a data unit.
- Click the **Display As** drop-down and select a display unit.
- Click the **Max Value Expression** to display the SNMP OID Browser where you select the OID that is the maximum value for the indicator type.
- Click the **Measured As** drop-down and select a data unit for the maximum value expression OID.
- Select the **Default allowed for new devices** check box to poll the indicator type by default when the object type is enabled and you enable the SNMP plugin for a device.
- In the **Note** field, enter any additional information you would like to include.
- Click **Save**.

> OID High Bits - SNMP version 1 specifications allow for 32 bit, unsigned integer counters. A 32 bit counter increments up to around four billion, then wraps back to zero, and continues. SNMP version 2 introduced 64 bit unsigned integer counters that can count much higher. 64 bit counters have twice the bits and twice the physical capacity of 32 bit counters to make them far more powerful and accurate. Manufacturers had two options to incorporate 64 bit counters.
>
> - Create a new 64 bit OID to represent the same thing as the 32 bit version.
> - Create a second 32 bit OID that represents the high bits of the 64 bit version.

17.3.15 VMware Poller

VMware object types are enabled by default. When you enable the VMware (Logical) and/or the VMware (Physical) topology sources on the Cluster Manager > **Cluster Settings** tab, these topology sources automatically discover VMware devices and enable the VMware plugin. This expedites the creation of Topology reports. For devices you manually add, the VMware plugin requires device specific configuration and you enable the VMware plugin for the device on the New Device page or on the Edit Device page.

17.3.15.1 VMware Object Types and Indicator Types

You cannot edit, add, or delete VMware object types or VMware indicator types. Discovery uses the object "objectTypeId" && ("name" || "name" == 'DiskIO' && "instance") to determine if a VMware object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **VMware Poller** to display the VMware object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click to display the VMware Indicator Type pop-up.
4. View the following VMware indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
    - **Default allowed for new devices** - Check box appears selected and the VMware plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for VMware poller object types.

17.3.16 Web Status Poller

The Web Status object type is enabled by default. There is one Web Status object type: Apache Mod Status. To poll Web Status devices, enable the device to send Web Status data to SevOne NMS and enable both the Web Status plugin and the SNMP plugin for the device on the New Device page or on the Edit Device page. The Authentication Settings page enables you to upload certificates to monitor Web Status via https.

17.3.16.1 Web Status Object Types and Indicator Types

You cannot add, edit, or delete Web Status object types or Web Status indicator types.

1. Click the **Filter** drop-down and select **Web Status Poller** to display the Apache Mod Status object type in the Object Types list.
2. Click on the **Apache Mod Status** object type to display its indicator types on the right.
3. Click  to display the Web Status Indicator Type pop-up.
4. View the following Web Status indicator type details.
    - **Indicator Name** - Displays the name of the indicator type.
    - **Description** - Displays the name to display.
    - **Indicator Type** - Displays the indicator type.
    - **Measure As** - Displays the indicator type data unit.
    - **Display As** - Displays the indicator type display unit.
    - **Maximum Value** - Check box appears selected when the indicator type has a maximum value.
    - **Default allowed for new devices** - Check box appears selected and the Web Status plugin polls the indicator type by default.
    - **Note** - Displays any additional information.
5. Click **Cancel**.

The Device Types section is irrelevant for Web Status poller object types.

17.3.17 WMI Poller

WMI object types are disabled by default. The WMI plugin discovers the WMI indicator types for the object types you enable. To poll WMI devices, enable the device to send WMI data to SevOne NMS, enable the WMI object types, and enable the WMI plugin for the device on the New Device page or on the Edit Device page.

17.3.17.1 WMI Object Types

Perform the following steps to manage WMI object types. Discovery uses the object "objectTypeId" && ("name" || "instanceName") to determine if a WMI object is a new object or an existing object with a new moniker.

1. Click the **Filter** drop-down and select **WMI Poller** to display the WMI object types in the Object Types list.
2. Click **Add** or click  to display the Add/Edit WMI Object Type pop-up.
3. In the **Name** field, enter the name of the object type.
4. In the **Class Name** field, enter the class name.
5. In the **Alias** field, enter the alias.
6. Select the **Enabled** check box to enable the WMI plugin to poll the object type.
7. Select the check box for the **Note** field to enable it for editing. Enter any additional information you would like to include.
8. Click **Save**.

17.3.17.2 WMI Atomic Indicator Types

Perform the following steps to manage WMI atomic indicator types. For WMI synthetic indicator types, please see section Synthetic Indicator Types below.

1. Click the **Filter** drop-down and select **WMI Poller** to display the WMI object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click **Add Atomic Indicator Type** or click  to display the Add/Edit WMI Indicator Type pop-up.
4. In the **Indicator Name** field enter the name of the indicator type.
5. In the **Description** field, enter the name to display.
6. Click the **Indicator Type** drop-down.
    - Select **Gauge** for indicators that have specific values when polled.

- Select **Counter32** for 32 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box to total the delta/differences between polls to provide the ability to graph things like the number of errors in a day, for example.
- Select **Counter64** for 64 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box.

7. Click the **Measure As** drop-down and select a data unit.
8. Click the **Display As** drop-down and select a display unit.
9. Select the **Maximum Value** check box to indicate the indicator type has a maximum value. You must select this check box if you want to use the indicators in this indicator type for percentile metrics.
10. Select the **Default allowed for new devices** check box to have the WMI plugin poll the indicator type by default when the object type is enabled and you enable the WMI plugin for a device.
11. In the **Note** field, enter any additional information you would like to include.
12. Click **Save**.

The Device Types section is irrelevant for WMI poller object types.

17.3.18 xStats

The xStats object types are enabled by default. The sources you define on the xStats Source Manager create xStats devices, object types, and indicator types. To monitor xStats data, contact your SevOne sales representative to discuss which xStats adapters are applicable for your implementation.

17.3.18.1 xStats Object Types

Perform the following steps to manage xStats object types.

1. Click the **Filter** drop-down and select **xStats** to display the xStats object types in the Object Types list.
2. Click **Add** or click  to display the Add/Edit xStats Object Type pop-up.
3. In the **Name** field, enter the name of the object type.
4. In the **Field Identifiers** field, enter the object type field identifiers.
5. Select the **Ignore** check box to have the xStats plugin not monitor the object type.
6. Click **Save**.

xStats Atomic Indicator Types

Perform the following steps to manage xStats atomic indicator types. For xStats synthetic indicator types, please see section Synthetic Indicator Types below.

1. Click the **Filter** drop-down and select **xStats** to display the xStats object types in the Object Types list.
2. Click on an object type to display its indicator types on the right.
3. Click **Add Atomic Indicator Type** or click  to display the Add/Edit xStats Indicator Type pop-up.
4. In the **Indicator Name** field enter the name of the indicator type.
5. In the **Description** field, enter the name to display.
6. Click the **Indicator Type** drop-down.

- Select **Gauge** for indicators that have specific values when polled.
- Select **Counter32** for 32 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box to total the delta/differences between polls to provide the ability to graph things like the number of errors in a day, for example.
- Select **Counter64** for 64 bit indicators that continue to increment. If you select this option, you can select the **Has Precalculated Deltas** check box.

- Click the **Measure As** drop-down and select a data unit.
- Click the **Display As** drop-down and select a display unit.
- Select the **Maximum Value** check box to indicate the indicator type has a maximum value. You must select this check box if you want to use the indicators in this indicator type for percentile metrics.
- In the **Field Identifiers** field, enter the object type field identifiers.
- Select the **Default allowed for new devices** check box to have the xStats plugin monitor the indicator type by default when the object type is enabled and you enable the xStats plugin for a device.
- Select the **Ignore** check box to have the xStats plugin ignore the indicator type.
- **Note** - Displays any additional information.
- Click **Save**.

The Device Types section is irrelevant for xStats poller object types.

17.4 Synthetic Indicator Types

Synthetic indicator types enable you to perform math on multiple metrics collected from multiple indicators on a single monitored object in order to calculate new KPIs.

Synthetic indicator types behave similar to the Calculation Editor that enables you to create custom calculation objects that perform math on multiple metrics collected from multiple objects and/or multiple synthetic indicators that you poll via the Calculation plugin. These features enable you create your own KPIs even when those KPIs do not exist on a device such as Percentage Loss, Percent Error, and Percent Idle. Synthetic indicators are always gauge type indicators and SevOne NMS evaluates all the indicator values the synthetic indicator uses as if the value is a gauge.

> **Example:** You want to monitor voice gateways to reveal which PRI gets the most or least usage. Typical poll metrics enable you to report on the status of individual bearer channels and not the sum of all channels at any given time. This makes it difficult to monitor and alert on total PRI usage. Synthetic indicators enable you to sum the bearer channel statuses (each channel gets a value of 1 when busy), divide by the total number of bearer channels (23), and then multiply by 100, to collect the desired metric for PRI % usage.

You can define synthetic indicator types for the following plugins: Calculation, Deferred Data, JMX, SNMP, WMI, and xStats.

1. Click the **Filter** drop-down and select one of the following: **Calculation Poller**, **Deferred Data**, **JMX Poller**, **SNMP Poller**, **WMI Poller**, or **xStats Poller** to display the object types in the list.
2. Click on an object type to display its indicator types on the right. If the object type does not have any indicator types, the Add Synthetic Indicator Type button does not appear.
3. Click **Add Synthetic Indicator Type** or click  next to a synthetic indicator type to display the Add/Edit Synthetic Indicator Type pop-up.
4. In the **Indicator Name** field, enter the name of the synthetic indicator type.
5. In the **Description** field, enter the name to display.
6. The **Synthetic Indicator Expression** field enables you to define the calculation.

   > If the border around the field turns red, your calculation is invalid and your graph results will be erroneous.

   > **Synthetic Indicator Expression :maxValue modifier**
   >
   > Each indicator in **Synthetic Indicator Expression** supports the **:maxValue** modifier. This modifier can be added to any indicator that has the maxValue. The user interface validates the modifier and approves it only if the underlying indicator has a maxValue set. Otherwise, the expression is marked as invalid and cannot be saved. For an expression that is invalid, you will get a pop-up message like **Modifier is not available for ${<indicator>:maxValue}** where **<indicator>** does not have a maxValue.
   > For example,
   > - **${ifHCInOctets:maxValue}** is a **valid** expression as ifHCInOctets has maxValue.
   > - **${SMDIMessageCountInbound24Hour:maxValue}** is an **invalid** expression as SMDIMessageCountInbound24Hour does not have maxValue.
   >
   > The discovered maxValue is retrieved and used for an indicator unless if the maxValue has been overwritten and in that case, the overwritten value is used.

   a. Click an indicator type in the **Available Source Indicators** field and drag it to the Synthetic Indicator Expression field. The Available Source Indicators field contains the indicator types associated to the object type you select in the hierarchy.
   b. Enter applicable operators in the Synthetic Indicator Expression field to formulate the calculation. Please refer to section Acceptable Operators below.
   c. Drag additional source indicator types and enter additional mathematical symbols to create the expression in the Synthetic Indicator Expression field.
7. The **Maximum Value Expression** field enables you to define the indicator type maximum value calculation.

   > If the border around the field turns red, your calculation is invalid and your graph results will be erroneous.

   > **Maximum Value Expression :maxValue modifier**

Each indicator in **Maximum Value Expression** supports the **:maxValue** modifier. This modifier can be added to any indicator that has the maxValue. The user interface validates the modifier and approves it only if the underlying indicator has a **maxValue** set. Otherwise, the expression is marked as invalid and cannot be saved. For an expression that is invalid, you will get a pop-up message like **Modifier is not available for ${<indicator>:maxValue}** where **<indicator>** does not have a maxValue.
For example,
- **${ifHCInOctets:maxValue}** is a **valid** expression as ifHCInOctets has maxValue.
- **${SMDIMessageCountInbound24Hour:maxValue}** is an **invalid** expression as SMDIMessageCountInbound24Hour does not have maxValue.

The discovered maxValue is retrieved and used for an indicator unless if the maxValue has been overwritten and in that case, the overwritten value is used.

   a. Click an indicator type in the **Available Source Indicators** field and drag it to the Maximum Value Expression field.
   b. Enter applicable operators in the Maximum Value Expression field to formulate the calculation. Please refer to section Acceptable Operators below.
   c. Drag additional source indicator types and enter additional mathematical symbols to create the expression in the Maximum Value Expression field.
8. Click the **Measure As** drop-down and select a data unit.
9. Click the **Display As** drop-down and select a display unit.
10. Select the **Default allowed for new devices** check box to have the plugin poll the indicator type by default when the object type is enabled and you enable the plugin for a device.
11. In the **Note** field, enter any additional information you would like to include.
12. Click **Save**.

17.4.1 Acceptable Operators

Your expression formula can include the following characters:

- **+** add
- **-** subtract
- ***** multiply
- **/** divide
- **&&** logical AND
- **||** logical OR
- **<=** less than or equal to
- **>=** greater than or equal to
- **!** not equal to
- **==** equal to
- **>** greater than
- **<** less than
- **^** raise x to the power of y
- **%** modulus
- **?:** if...then...else
- **isnan** is Not a Number. This evaluates to 1 if the value is not a number. Otherwise, it evaluates to 0.
- **isValid** is valid. This evaluates to 1 if the value has been discovered and is not **isnan**. Otherwise, it evaluates to 0.
- **useIfValid** use if valid. This evaluates to the value if it has been discovered and is not **isnan**. It evaluates to the second argument otherwise.

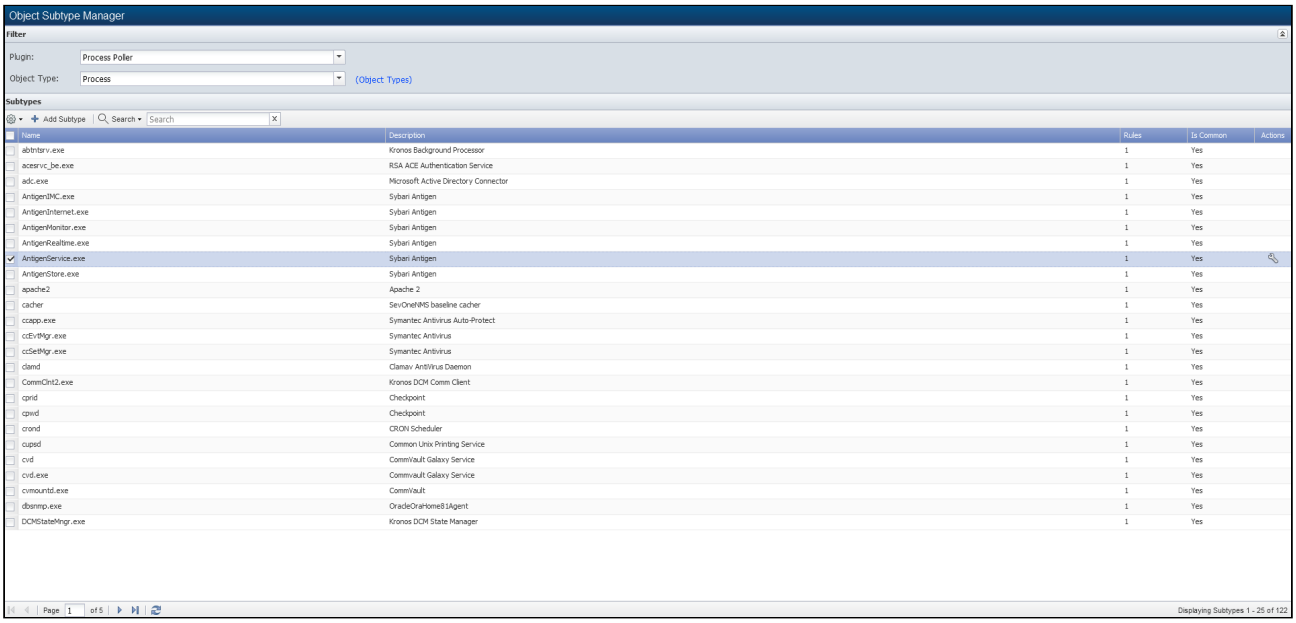
If your calculation results in either of the following invalid values, there will be a gap in your graph: Not a Number (NAN) and Infinity (+/-INF). The following is how SevOne NMS attempts to prevent invalid values.
In sequence of processing:

- Zero divided by zero results in NAN.
- Any positive value divided by zero results in +INF.
- Any negative value divided by zero results in -INF.
- Zero multiplied by +/-INF results in NAN.
- Any value added to, subtracted from, multiplied by, divided by, or divided from NAN results in NAN.
- Any value compared to NAN (<, <=, ==, >=, >) results in 0. NAN != NAN.
- Any value compared to +INF is less than +INF, except that +INF == +INF
- Any value compared to -INF is greater than -INF, except that -INF == -INF
- Any value added to or subtracted from +INF results in +INF
- Any positive value multiplied by +/-INF results in +/-INF
- Any value divided by +/-INF results in 0

18 Object Subtype Manager

The Object Subtype Manager enables you to manage the object subtypes that group the objects you want the Deferred Data plugin, Process plugin, and SNMP plugin to poll. Object discovery varies from plugin to plugin. The Object Rules page enables you to define rules to disable polling of objects and the Object Manager enables you to manage the objects on each device.

To access the Object Subtype Manager from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **Object Subtype Manager.** You can also access the Object Subtype Manager from the Object Types page.



18.1 Object Subtypes List

The list displays the object subtypes for the object type you select above the list. When the object type is disabled, no metrics are polled for the object subtypes you define.

- **Name** - Displays the object subtype name.
- **Description** - Displays the object subtype description.
- **Rules** - Displays the number of object subtype rules.
- **Is Common** - Displays *Yes* when you mark the object subtype as common. Displays *No* when you do not mark the object subtype as common. Common is a filter criterion when you define a policy and TopN views.

18.2 Manage Object Subtypes

Perform the following steps to manage object subtypes. Filters enable you to display object subtypes in the list.

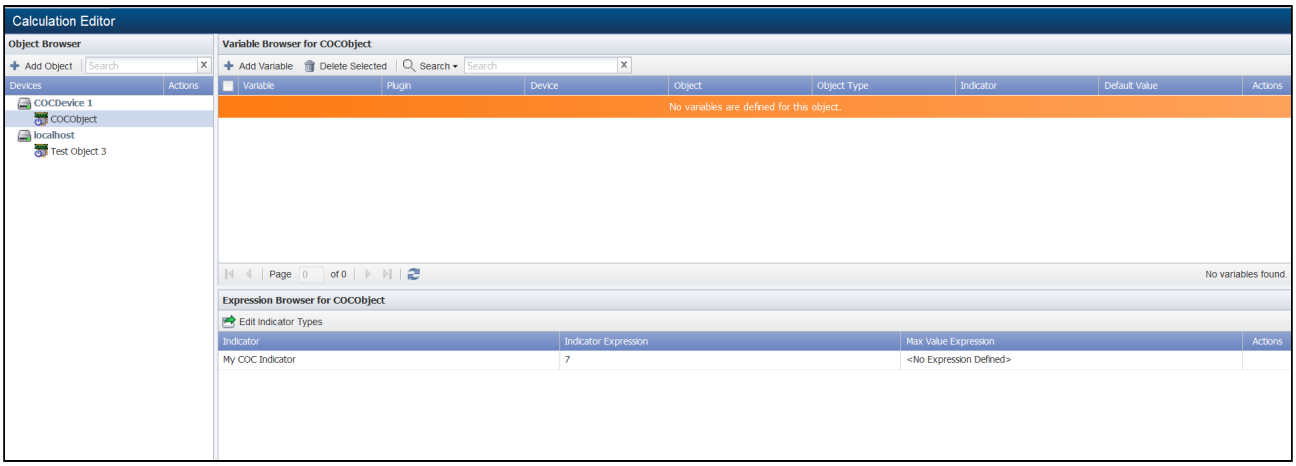
1. Click the **Plugin** drop-down and select the plugin whose object subtypes you want to manage.
2. Click the **Object Type** drop-down and select the object type that contains the object subtypes to manage. Click the Object Types link to navigate to the Object Types page.
3. Click **Add Subtype** or click  to display the Add/Edit Object Subtype pop-up.
4. In the **Name** field, enter the name of the object subtype.
5. In the **Description** field, enter the object subtype description.
6. Select the **Is Common** check box to enable the objects associated with the subtype to appear in lists of common objects.
7. For the Process plugin and for the SNMP plugin, click **Add** to add a new line to the Rules list.
    - For the Process plugin:
        i. In the **Run Name** field, enter the run name.
        ii. In the **Run Path** field, enter the run path.
        iii. In the **Run Argument** field, enter the run argument.
    - For the SNMP plugin:
        - In the **Identifier** field, enter the string to match in order to apply the rule.
8. Click **Update** to save the rule.
9. Repeat the Add rule steps to define additional rules. Each rule is evaluated and appropriate rules are applied.

10. Click **Save**.

19 Calculation Editor

The Calculation Editor enables you to define objects that use the calculations of polled indicator data you define as variables. Variable calculations can combine data SevOne NMS polls from the indicators across multiple objects on multiple devices.

To access the Calculation Editor from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **Calculation Editor**.



19.1 Prerequisites

- Enable the Calculation plugin for devices on the **New Device** page and/or the **Edit Device** page. For details, please refer to sections **Calculation Plugin**, **Edit Device**, and **New Device** in *SevOne NMS User Guide*.
- Define calculation object types and indicator types on the Object Types page.

19.2 Object Browser

The Object Browser section on the left displays a list of the devices for which you enable the Calculation plugin and for which you define calculation objects. As you define calculation objects, additional devices appear in the list.

1. If needed, click on a device in the list to display the calculation objects the Calculation plugin polls on the device.
2. Click **Add Object** or click  to display the Add/Edit Object pop-up.
3. For a new object, click the **Parent Device** drop-down and select the device on which to discover and poll the object. This list displays all devices for which you enable the Calculation plugin. You cannot edit this field after you save an object.
4. For a new object, click the **Object Type** drop-down and select a Calculation plugin object type. To add an object type to the list, click  to navigate to the Object Types page where you can manage the Calculation plugin object types. You cannot edit this field after you save an object.
5. In the **Object Name** field, enter the name of the Calculation plugin object. This name cannot begin or end with spaces and must include at least one character.
6. In the **Object Description** field, enter the object description.
7. In the **Default Value** field, enter a numeric value to use in the calculation should the object type not poll data for the time span of the report.
8. Click **Save**.
9. After you save your new object, the device that you added it to will appear in the **Search** field. Your new object is automatically selected in the Object Browser, making it easy to locate.

19.3 Variable Browser

Click on an object in the Object Browser list to display its variables in the Variable Browser section and expression in the Expression Browser section on the right.

1. Click **Add Variable** or click  to display the Add/Edit Calculation Variable pop-up.
2. In the **Variable Name** field, enter the name of the variable for the object to use.
3. Click the **Plugin** drop-down and select the plugin that polls the object from which to calculate data.
4. Click the **Device** drop-down and select the device from which the data to calculate is polled.
5. Click the **Object** drop-down and select the object whose indicator data is to be calculated.

6. Click the **Indicator** drop-down and select the indicator whose data is to be used in the calculation.
7. Select the **Default Value Null** check box to make the value default to null if there is no poll data for the variable during the report time span. Leave clear and enter a value to use a specific value for time spans with no poll data.
8. Click **Save**.

19.4 Expression Browser

Click on an object in the Object Browser section to display its variables in the Variable Browser section and indicators in the Expression Browser section on the right.

1. Click **Edit Indicator Types** to navigate to the Object Types page where you can add and edit indicator types.
2. Click  next to an indicator to display the Expression Editor pop-up.
3. In the **Source Variable** field, click a variable and drag it to the **Indicator Expression** field.
4. Enter applicable operators in the Indicator Expression field to formulate the calculation. Please see section Acceptable Operators below.
5. Drag additional source variables and enter additional mathematical symbols to create the expression in the Indicator Expression field.
6. Repeat the previous steps in the **Maximum Value Expression** field to define a maximum value expression.

   Maximum Value Expression uses the maximum value of the chosen indicator for each variable being referenced in the expression. Let's say the variables are set as shown in the table below.

Variable	Plugin	Device	Object	Object Type	Indicator
${v1}	SNMP Poller	D1	ens160	Interface	HC In Octets
${v2}	SNMP Poller	D1	ens160	Interface	HC Out Octets
${v3}	SNMP Poller	D2	ens160	Interface	HC In Octets
${v4}	SNMP Poller	D2	ens160	Interface	HC Out Octets

   If **Maximum Value Expression** is **${v1}+${v2}+${v3}+${v4}** then, it will take the maximum value of the indicator chosen for each of these variables. i.e., it will use the maximum value of the following indicators to evaluate maximum value expression.
   - **HC In Octets** for Device **D1**
   - **HC Out Octets** for Device **D1**
   - **HC In Octets** for Device **D2**
   - **HC Out Octets** for Device **D2**
7. Click **Save**.

19.4.1 Acceptable Operators

Your expression formula can include the following characters:

If the border around the field turns red, your calculation is invalid and your graph results will be erroneous.

- **+** add
- **-** subtract
- ***** multiply
- **/** divide
- **&&** logical AND
- **||** logical OR
- **<=** less than or equal to
- **>=** greater than or equal to
- **!** not equal to
- **==** equal to
- **>** greater than
- **<** less than
- **^** raise x to the power of y
- **%** modulus

- **?:** if...then...else

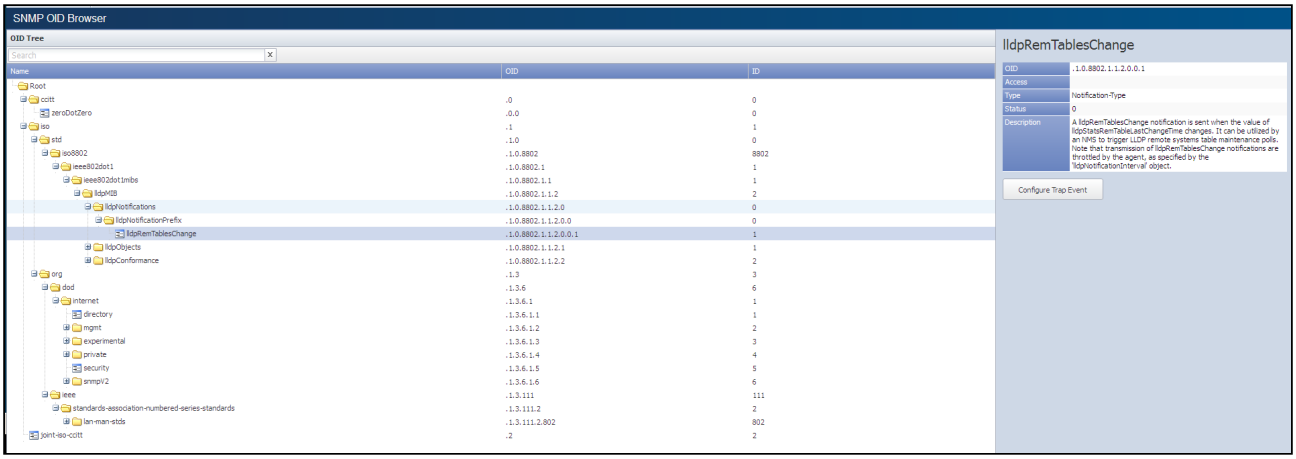
If your calculation results in either of the following invalid values, there will be a gap in your graph: Not a Number (NAN) and Infinity (+/-INF). The following is how SevOne NMS attempts to prevent invalid values.
In sequence of processing:

- Zero divided by zero results in NAN.
- Any positive value divided by zero results in +INF.
- Any negative value divided by zero results in -INF.
- Zero multiplied by +/-INF results in NAN.
- Any value added to, subtracted from, multiplied by, divided by, or divided from NAN results in NAN.
- Any value compared to NAN (<, <=, ==, >=, >) results in 0. NAN != NAN.
- Any value compared to +INF is less than +INF, except that +INF == +INF
- Any value compared to -INF is greater than -INF, except that -INF == -INF
- Any value added to or subtracted from +INF results in +INF
- Any positive value multiplied by +/-INF results in +/-INF
- Any value divided by +/-INF results in 0

20 SNMP OID Browser

The SNMP OID Browser enables you to select the SNMP Management Information Bases (MIB) object identifiers (OID) you use to create SNMP object types and SNMP traps. MIBs are the files that enable the raw machine generated OIDs to display in a way that is more understandable to users. The MIB Manager enables you to add and manage MIBs. You can contact your SevOne Support Engineer to perform device certifications to add additional MIBs.

You can access the SNMP OID Browser from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **SNMP OID Browser**. Typically you access the SNMP OID Browser from an SNMP object type definition workflow or from a trap event configuration workflow. When an applicable OID appears on the right, click **Select OID** to return to the Object Types page or click **Configure Trap Event** to navigate to the Trap Event Editor.



20.1 OID Tree and OID Information

> **Search** field allows you to filter the list of OIDs by name or by number.

The OID Tree section displays the OID hierarchical structure. Navigate the OID tree hierarchy and select an OID to display more information on the right.

When you select an OID that is an actual trap that could be sent to SevOne NMS, the **Configure Trap Event** button is available to enable you to use the OID for the trap event. When you access the SNMP OID Browser from the Object Types page the **Select OID** button enables you to associate the OID with the SNMP Object Type or the SNMP Indicator Type.

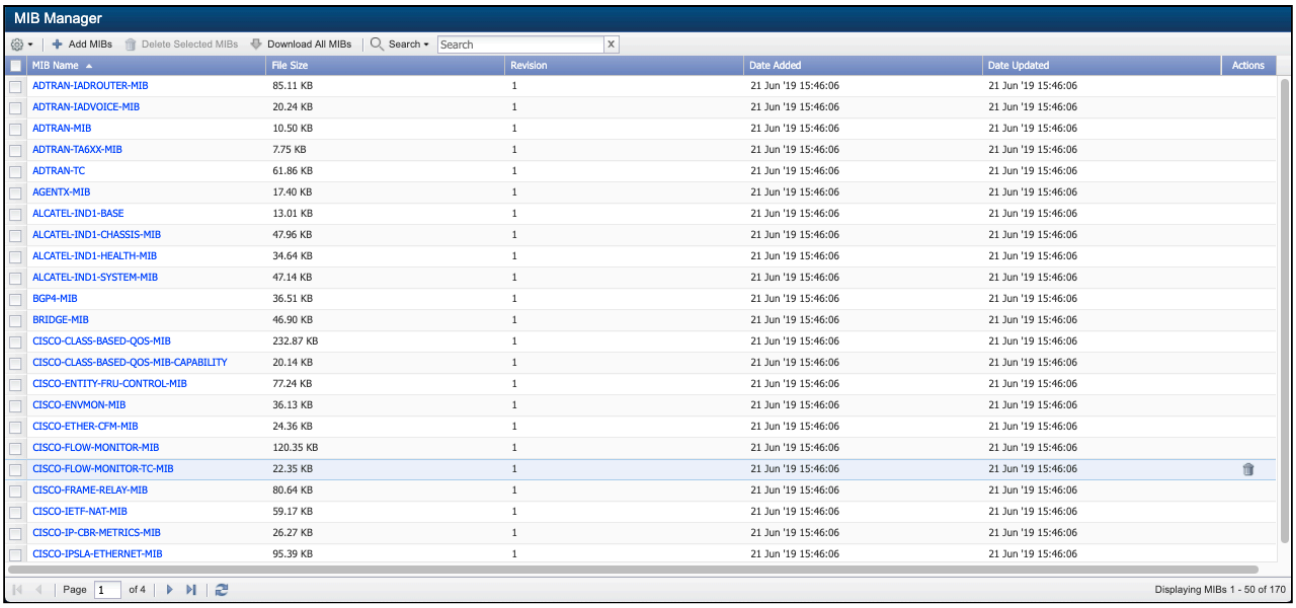
The OID Information section displays the following information.

- **Name** - displays the OID name.
- **OID** - displays the OID number.
- **Access** - displays the type of access available for the OID such as *Read*, *Read/Write*, etc.
- **Type** - displays how the OID appears such as *String*, *Integer*, etc.
- **Status** - displays the OID status such as *Current*, *Deprecated*, or *0* (no status).
- **Description** - displays the OID description.

21 MIB Manager

The MIB Manager enables you to view MIB details and to add MIBs. MIBs are the files that enable the raw machine generated OIDs to display in a way that is more understandable to users. SevOne NMS provides a list of standard MIBs.

To access the MIB Manager from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **MIB Manager**.



21.1 MIB List

The list displays the following MIB information.

- **MIB Name** - displays the MIB name. Click the name to view MIB details.
- **File Size** - displays the size of the MIB file.
- **Revision** - displays the number of times you upload the MIB to SevOne NMS.
- **Date Added** - displays the date the MIB was added.
- **Date Updated** - displays the date the MIB was most recently edited.

21.1.1 Manage MIBs

Perform the following steps to manage MIBs.

- To download some MIBs, select the check box for each MIB to download, click  , and select **Download Selected MIBs**.
- Click **Add MIBs** to display the **MIB Uploader** pop-up.

  a. Click **Select MIB(s)** to display the File Upload pop-up.
  b. Navigate to the MIB file to upload and select it. To add multiple MIBs, zip the MIB files into a .zip format file and select the .zip file.
  c. Click **Open** on the File Upload pop-up.
  d. Click **Upload** on the MIB Uploader pop-up.

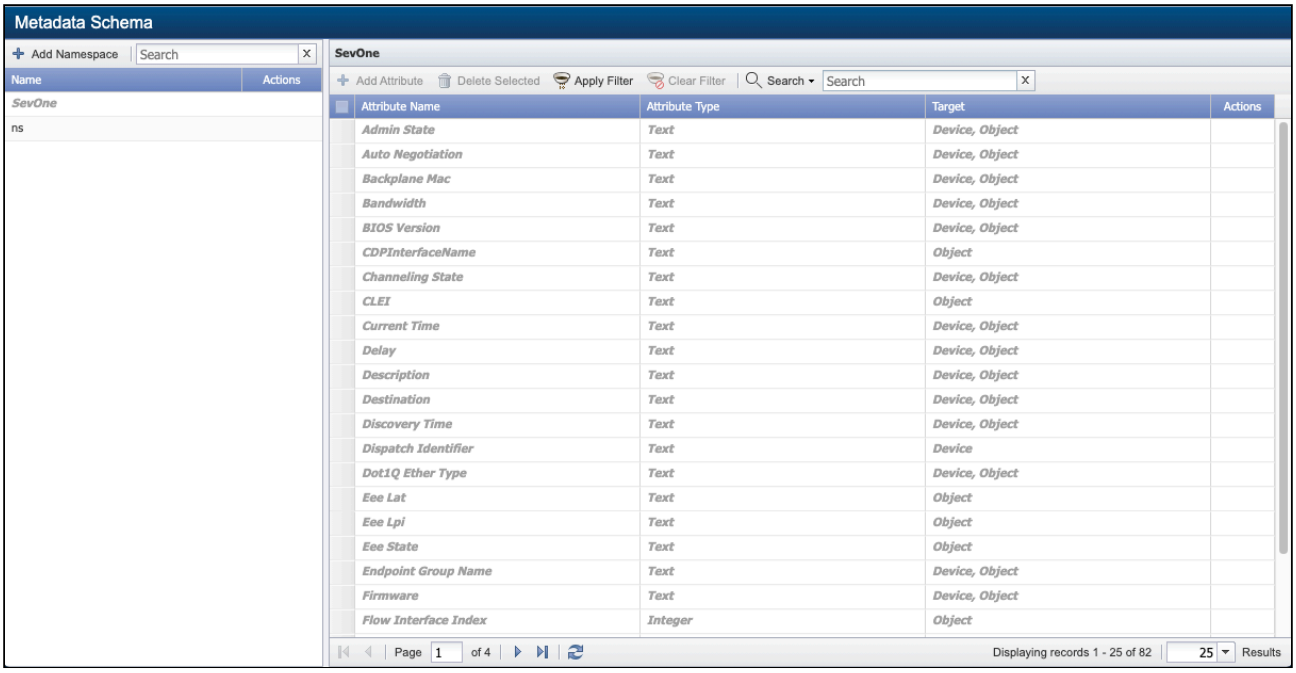
> It may take up to 30 minutes for the change to take effect.

- Click **Delete Selected MIBs** to remove one or more MIBs selected. You may also right-click on a row and click **Delete** to delete the MIB on the row you are on.
- Click **Download All MIBs** to create a .zip file of all MIBs that you can download for email, backup, etc. You may also right-click on a row and click **Download** to only download the MIB on the row that you are on.
- Click a name in the **MIB Name** column to view the MIB details for the MIB name selected.

22 Metadata Schema

The Metadata Schema page enables you to manage metadata attributes that are specific to your network. SevOne NMS metadata attributes display in italic font. You cannot edit or delete SevOne NMS metadata attributes.

To access the Metadata Schema page from the navigation bar, click the **Administration** menu and select **Metadata Schema**.



22.1 Metadata List Filters

Filters enable you to limit the metadata that appear in the list. All filters are optional and cumulative.

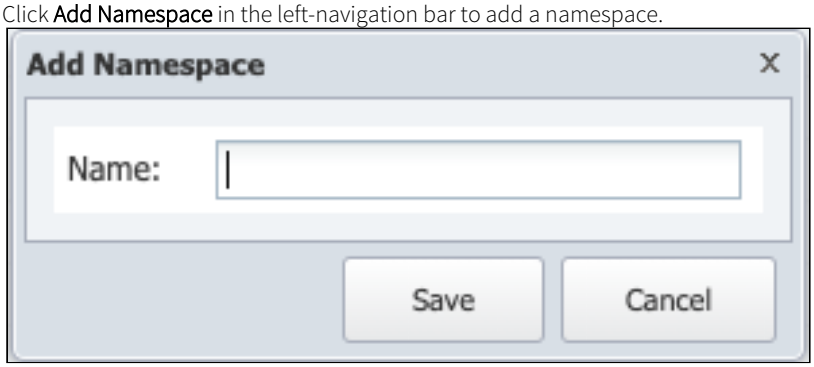
1. Select a namespace from the left-navigation bar.
2. Click **Apply Filter** to display the **Filter Options** pop-up.
3. Click the **Target** drop-down and select a target.
4. Click the **Type** drop-down and select a type.
5. Click **Apply**.

22.2 Add / Edit Namespaces

Namespace(s) must be created before you can define the attributes.

22.2.1 Add Namespace

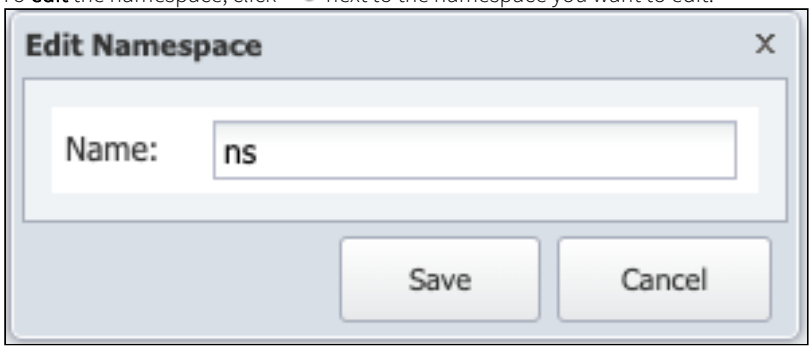
1. Click **Add Namespace** in the left-navigation bar to add a namespace.



2. In **Add Namespace** pop-up, enter a name for the namespace in field **Name**.
3. Click **Save**.

22.2.2 Edit Namespace

1. To **edit** the namespace, click  next to the namespace you want to edit.



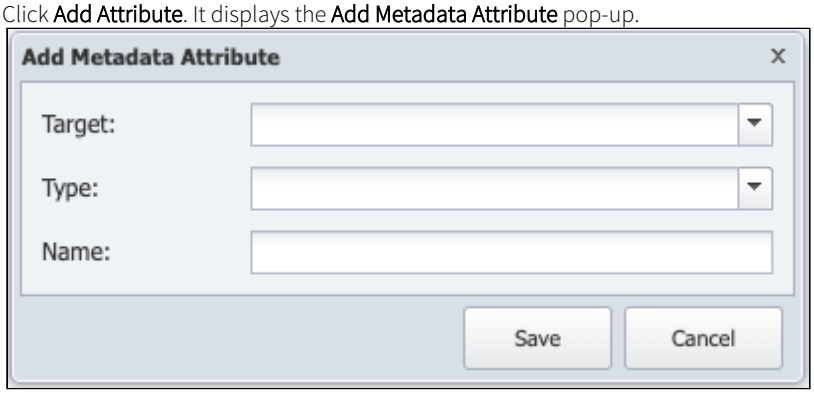
2. **Edit Namespace** pop-up appears.
3. In the **Edit Namespace** pop-up, enter the namespace name you want to change it to in field **Name**.
4. Click **Save**.

22.3 Add / Edit Attributes

Each metadata attribute is grouped by namespace and can have multiple targets. The attribute target provides the ability to edit the value for each attribute. The attribute type determines what data the attribute presents.

22.3.1 Add Attributes

1. Select a namespace in the left-navigation bar to which you want to add the attributes.
2. Click **Add Attribute**. It displays the **Add Metadata Attribute** pop-up.



3. **Target** - refers to the areas in SevOne NMS that you can apply a specific metadata attribute to. . Each metadata attribute can have multiple targets. Click the Target drop-down.

> **Example**
>
> Assume that your company has several offices throughout the United States. One of your offices is in San Diego, and you want to provide a site contact phone number for all of the *devices* at the San Diego location. In this case, you will select **Device** as a target. This gives you the ability to use your site contact phone number attribute for individual *devices*.
> Besides **Device**, you can also select **Object**, **Device Group**, **Object Group**, **Object Type**, and **Indicator Type** as targets.

   a. Select **Device** to enable the association of values from the **Device Manager** for specific devices.
   b. Select **Object** to enable the association of values from the **Object Manager** for specific objects.
   c. Select **Device Group** to enable the association of values from the **Device Groups** page for specific device groups and from the Device Types page for specific device types.
   d. Select **Object Group** to enable the association of values from the Object Groups page for specific object groups.

e. Select **Object Type** to enable the association of values from the Object Types for specific object types.
f. Select **Indicator Type** to enable the association of values for specific indicator types.

> The following sections can be found in *SevOne NMS System Administration Guide*.
> - Device Types
> - Object Groups
> - Object Types
>
> The following sections can be found in *SevOne NMS User Guide*.
> - Device Groups
> - Device Manager
> - Object Manager

4. **Type** - is the kind of data the attribute presents. Click the Type drop-down.

> Type is used to validate entry inputs. It specifies the format of the data that users will provide for a specific attribute. If you would like to include the installation date for a device, for example, you may create a metadata attribute and call it **Installation Date**. Because you want it to be in date format, you would select the attribute type **Date/Time**. This means that data provided for the Installation Date attribute must conform to the Date/Time format.
> Additional attribute types include **IP Address**, **MAC Address**, **Integer**, **Latitude and Longitude**, **Regular Expression**, and **URL**. There is also an attribute type called **Text**, which you can use for any number of things, such as phone numbers, names of people, serial numbers, notes, etc.

a. Select **Date/Time** to enable the value to be in a date/time format.
b. Select **Integer** to enable the value to be numeric.
c. Select **IP Address** to enable the value to be an IP address.
d. Select **Latitude and Longitude** to enable the value to be a latitude and longitude.
e. Select **MAC Address** to enable the value to be a MAC address.
f. Select **Text (Validated)** to enable the entry of a regular expression that is validated for accuracy. A **Regular Expression** field appears to enable you to enter the regular expression on which to validate the attribute values.

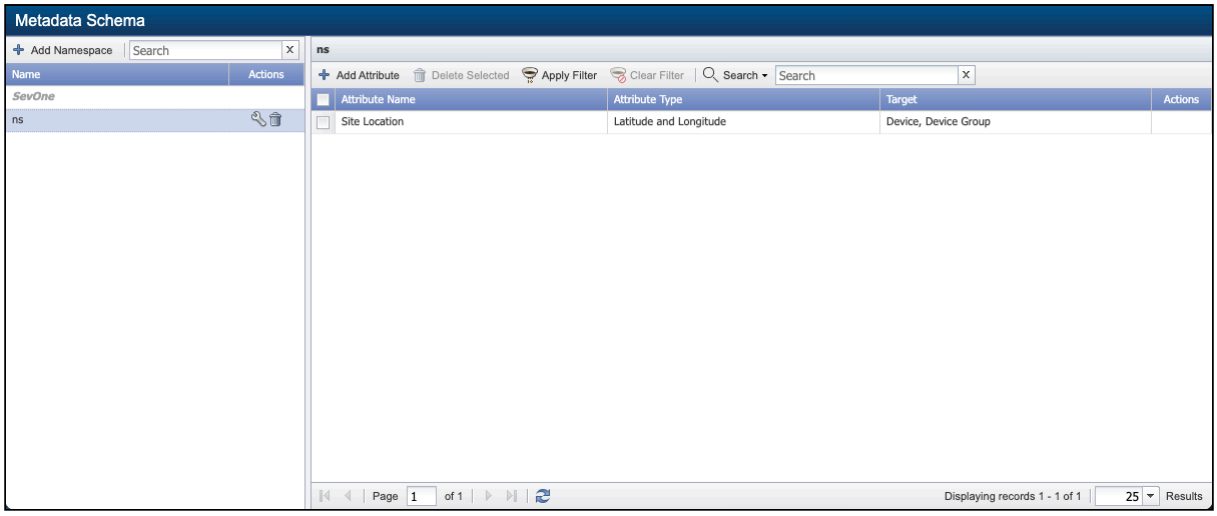
> The input for the Regular Expression field needs to be a full regular expression with delimiters and modifiers.

g. Select **Text** to enable the value to be text.
h. Select **URL** to present the value as a clickable link to a valid URL.

5. **Name** - is the name of the metadata attribute.

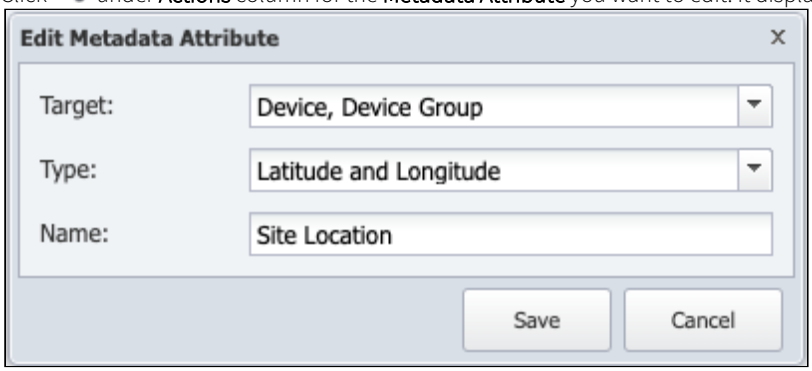
> For example, if you have an attribute for providing the location of a *device*, you may name it **Site Location**.

6. Click **Save**.



22.3.2 Edit Attributes

1. Select a namespace in the left-navigation bar for which you want to edit the attributes.
2. Click under **Actions** column for the **Metadata Attribute** you want to edit. It displays **Edit Metadata Attribute** pop-up.

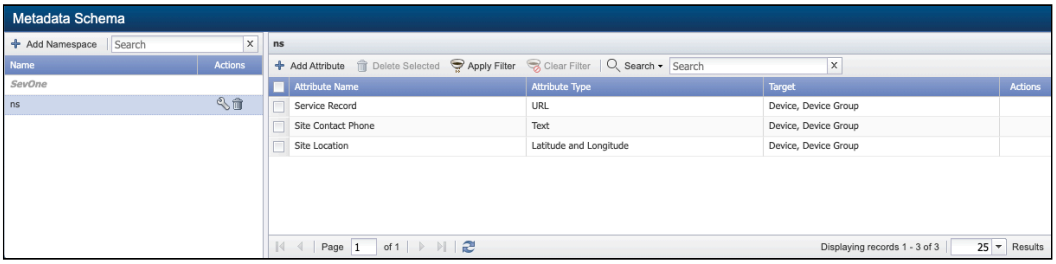


Please refer to Metadata Attributes for details on the fields you want to edit.

3. Click **Save** after you have modified the metadata attribute(s).

- To add additional attributes to a namespace, please refer to section Add Attributes.

Additional attributes Site Location, Site Contact Phone, and Service Record added to namespace 'ns'

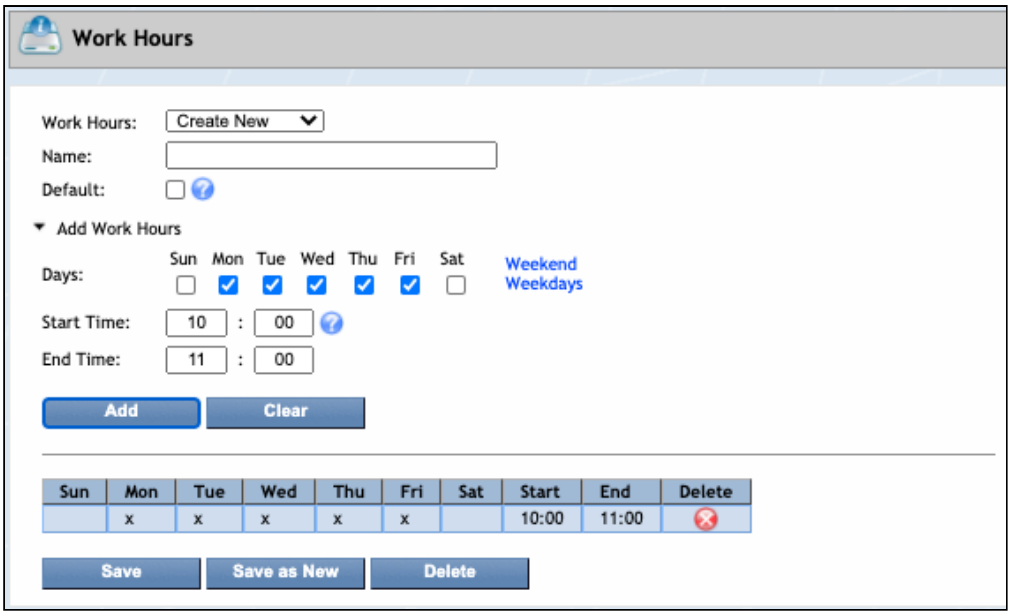


- To edit existing attributes in a namespace, please refer to section Edit Attributes.

23 Work Hours

The Work Hours page enables you to create work hours for devices. Work hours enable you to create reports that present statistics for specific work hours. Pages that use work hours include: Report Attachment Wizard, Instant Graphs, TopN Reports, New Device, Edit Device, and Device Manager.

To access the Work Hours page from the navigation bar, click the **Administration** menu and select **Work Hours**.



23.1 Manage Work Hours

Enter times in a 24 hour format. A day starts at 00:00 (12:00 AM) and ends at 23:59 (11:59 PM). For an example of a time span that crosses midnight such as 5 PM Monday to 2 AM Tuesday: Enter one span Monday 17:00-23:59 and enter a second time span Tuesday 00:00 - 2:00.

1. Click the **Work Hours** drop-down and select **Create New** or select a work hour group to edit.
2. In the **Name** field, enter the name of the work hours group.
3. Select the **Default** check box to have this work hours group appear as the default work hours group for all new devices. You can designate one default work hours group.
4. Click the **Add Work Hours** to display day and time entry fields.
5. In the **Days** section, select the check box for each day to include in the work hours group.
6. In the **Start Time** fields, enter the start time, in 24 hour format.
7. In the **End Time** fields, enter the end time, in 24 hour format.
8. Click **Add** to add the work hours group to the list.
9. Repeat these steps to add multiple work hours groups.
10. Click **Save** to update the work hours group with the changes or click **Save as New** to create a copy of the work hours group.

24 Enable JMX

Java Management Extensions (JMX) is a Java specification technology (defined in JSR-160) that provides a standard means for Java applications to publish indicators to JMX compliant management and monitoring systems.

This topic describes how to enable JMX devices to send JMX data to SevOne NMS. This workflow is outside of the SevOne NMS application and may not present all of the steps your network requires to enable devices to send JMX data. If the following instructions are not applicable for your network please reference the device manufacturer's documentation.

Related SevOne NMS workflows include the following.

- The Device Manager provides access to the New Device page and the Edit Device page where you enable the JMX plugin for a device.
- The Object Types page enables you to enable or disable the JMX object types and indicator types you want the JMX plugin to poll in your network.
- The Indicator Type Map page enables you to enable or disable the device specific indicators you want the JMX plugin to poll.

24.1 Send JMX Data to SevOne NMS

Enter the following command on the JMX device.

```
Hostname -i
```

The result should not equal 127.0.0.1

24.1.1 Tomcat

Enter the following commands for the Tomcat Startup script.

```
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.port=8007"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.authenticate=false"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.ssl=false"
JAVA_OPTS="$JAVA_OPTS -Djava.rmi.server.hostname=192.168.50.213"
```

WebLogic

Enter the following commands for the WebLogic Startup script.

```
/opt/Oracle/Middleware/wlserver_10.3/samples/domains/wl_server/bin/
startWebLogic.sh
JAVA_OPTIONS="$JAVA_OPTIONS -Dcom.sun.management.jmxremote"
JAVA_OPTIONS="$JAVA_OPTIONS -Dcom.sun.management.jmxremote.port=8007"
JAVA_OPTIONS="$JAVA_OPTIONS
-Dcom.sun.management.jmxremote.authenticate=false"
JAVA_OPTIONS="$JAVA_OPTIONS -Dcom.sun.management.jmxremote.ssl=false"
JAVA_OPTIONS="$JAVA_OPTIONS -Djava.rmi.server.hostname=192.168.30.251"
JAVA_OPTIONS="$JAVA_OPTIONS
-Djavax.management.builder.initial=weblogic.management.jmx.mbeanserver.WLSMB
eanServerBuilder"
```

You must also enable this on the Administrative page.

```
domain -> Configuration -> general (Advanced) , usePlatformMBean ,
exportPlatformMBean
```

24.1.2 JBoss

Enter the following commands to enable JBoss to send JMX data. Change the hostname and IP to /etc/hosts to connect. The servername cannot be listed under 127.0.0.1

In `/opt/jboss/bin/run.sh` add `#Setup jmx remoting`

```
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.authenticate=false"
```

```
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.ssl=false"
```

```
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote.port=8007"
```

Use the JBoss MBeanServerBuilder.

```
JAVA_OPTS="$JAVA_OPTS -Djboss.platform.mbeanserver"
```

```
JAVA_OPTS="$JAVA_OPTS
-Djavax.management.builder.initial=org.jboss.system.server.jmx.MBeanServerBu
ilderImpl"
```

Use the jboss logmanager.

```
JAVA_OPTS="$JAVA_OPTS
-Djava.util.logging.manager=org.jboss.logmanager.LogManager"
```

```
JAVA_OPTS="$JAVA_OPTS
-Dorg.jboss.logging.Logger.pluginClass=org.jboss.logging.logmanager.LoggerPl
uginImpl"
```

```
JBOSS_CLASSPATH="../lib/jboss-logmanager.jar"
```

24.1.3 GlassFish

Increase the monitoring to HIGH on the Web Admin page and enter the following commands to enable GlassFish v 3.1 to send JMX data.

In the domain configuration file `/glassfish/domains/domain1/config/domain.xml` find `<java-config>` and add:

```
-Dcom.sun.management.jmxremote.ssl=false
```

```
-Dcom.sun.management.jmxremote.authenticate=false
```

Then find `<admin-service>` and add/modify:

```
<jmx-connector accept-all="true" port="8007" address="0.0.0.0" security-
enabled="false" name="system" auth-realm-name="admin-realm" />
```

24.1.4 WebSphere 6.1

Enter the following commands for WebSphere 6.1.

```
Install the server { yum install compat-libstdc++-* libXp libXmu }
```

```
Run the server { /etc/init.d/webspherejmxserverNode01_was.init start }
```

Log in via the admin console and make sure the JMX RMI connection is established.

```
Use
the Python script ListPorts to get the admin port. { ./wsadmin.sh
-conntype none -lang jython -profileName AppSrv01 -f /opt/ListPorts.py }
```

```
Log into the server { admin:admin }
```

Back to the CLI, disable security.

```
{
```

```
    ./wsadmin.sh -conntype NONE
    wsadmin>securityoff
    wsadmin>$AdminConfig save
}
```

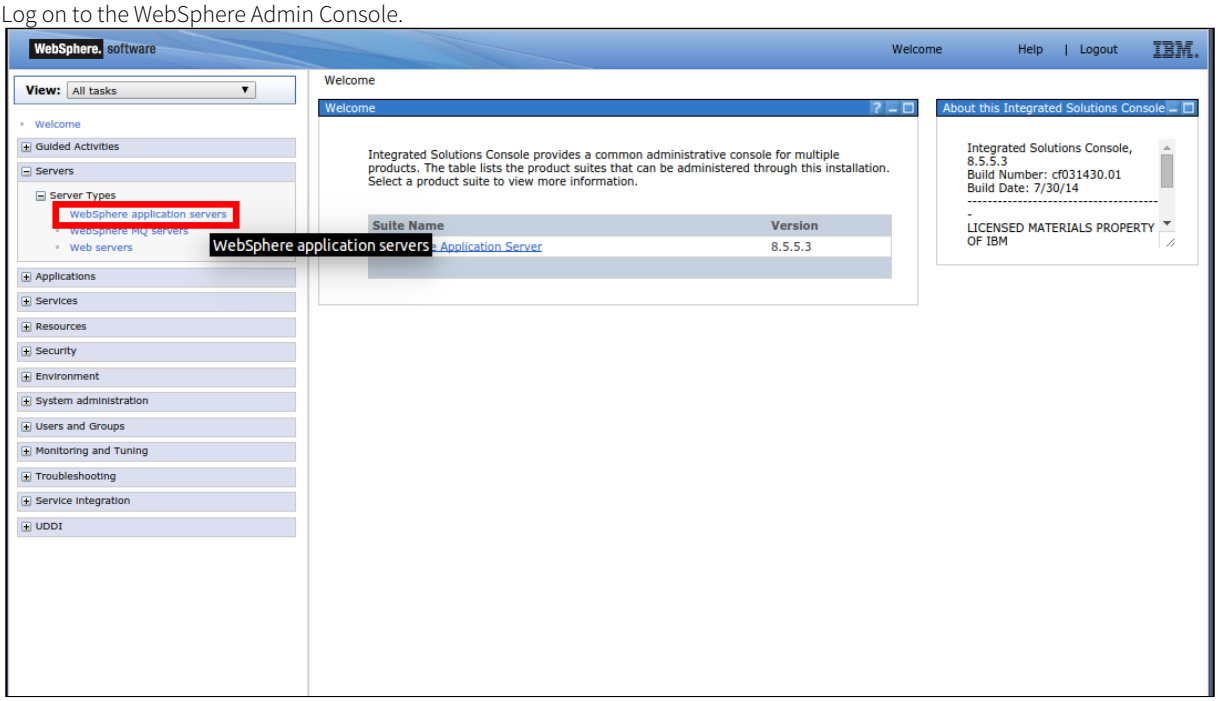
Connect with the following script.

```
#!/bin/bash
HOST=
PORT=
WAS_HOME=/home/dkozlowski/IBM/WebSphere/AppServer
CLIENTSAS="-Dcom.ibm.CORBA.ConfigURL=file:`pwd`/sas.client.props"
PROVIDER="-Djava.naming.provider.url=corbaname:iiop:$HOST:$PORT"
PROPS=
#PROPS="$PROPS $CLIENTSAS"
#PROPS="$PROPS $PROVIDER"
CLASSPATH=
CLASSPATH="$CLASSPATH:$WAS_HOME/java/lib/tools.jar"
CLASSPATH="$CLASSPATH:$WAS_HOME/runtimes/com.ibm.ws.admin.client_7.0.0.jar"
CLASSPATH="$CLASSPATH:$WAS HOME/runtimes/
com.ibm.ws.ejb.thinclient_7.0.0.jar"
CLASSPATH="$CLASSPATH:$WAS_HOME/runtimes/com.ibm.ws.orb_7.0.0.jar"
CLASSPATH="$CLASSPATH:$WAS_HOME/java/lib/jconsole.jar"
URL=service:jmx:iiop://$HOST:$PORT/jndi/JMXConnector
java -classpath $CLASSPATH $PROPS sun.tools.jconsole.JConsole $URL
```

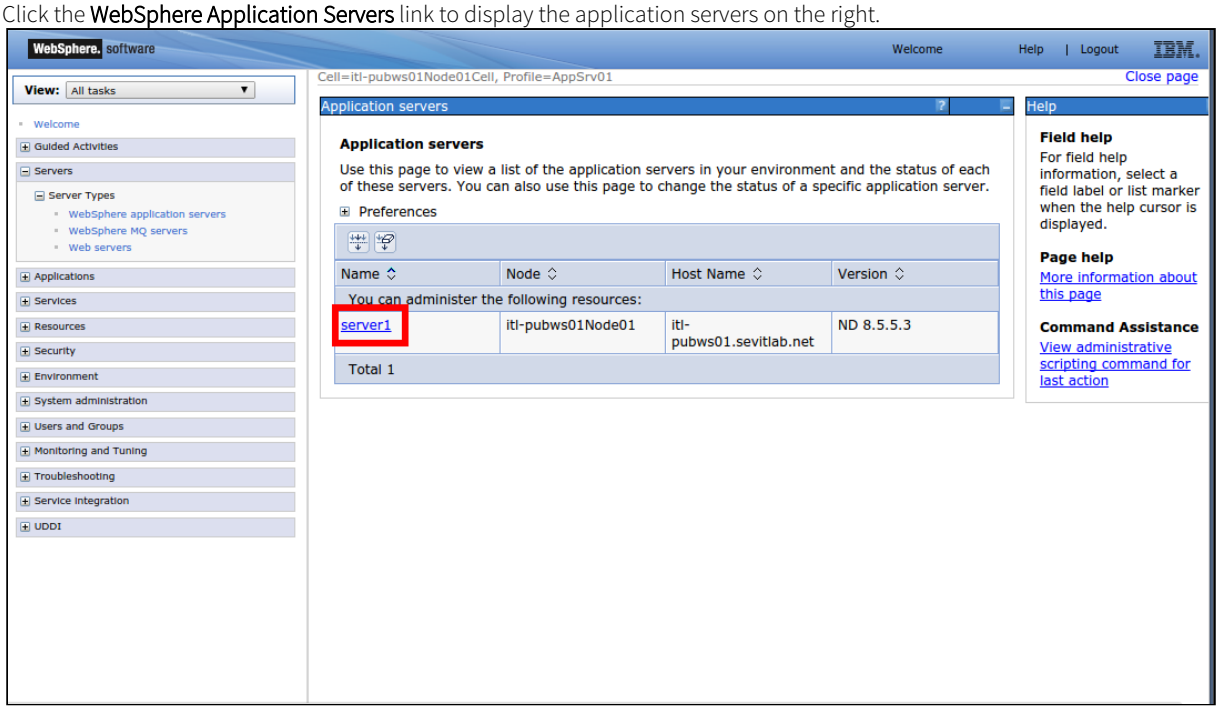
24.1.5 WebSphere 8.5

WebSphere JMX Configuration

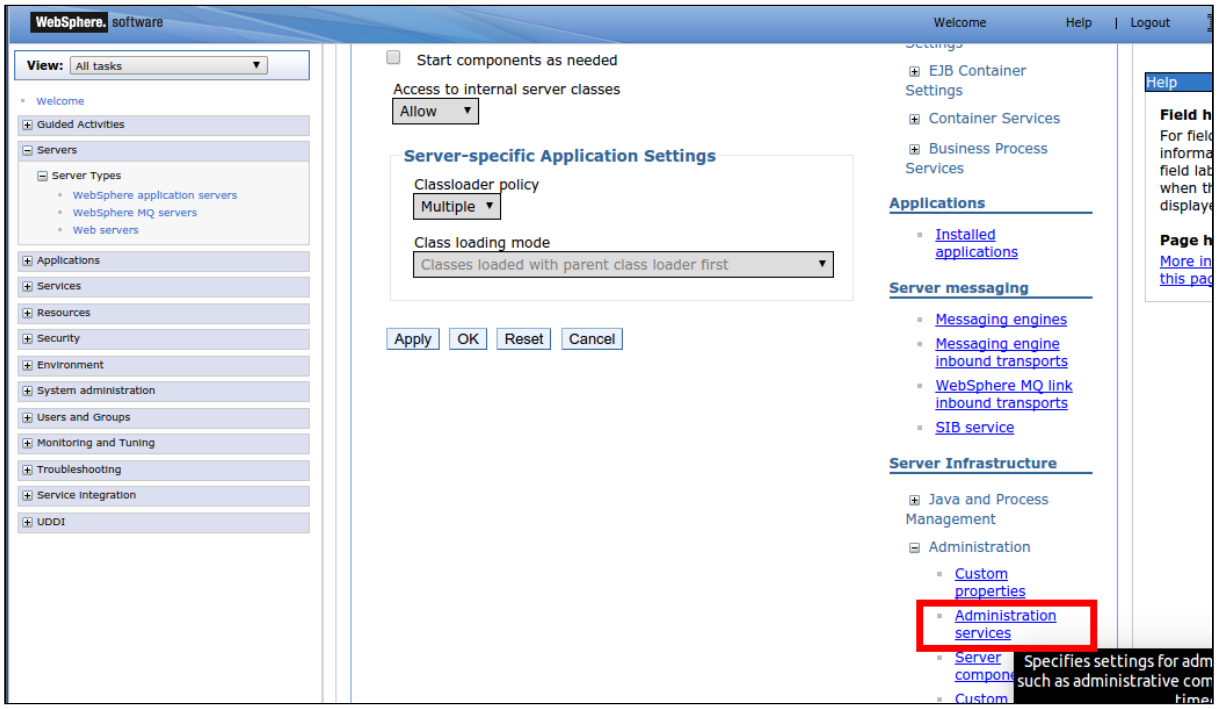
1. Log on to the WebSphere Admin Console.



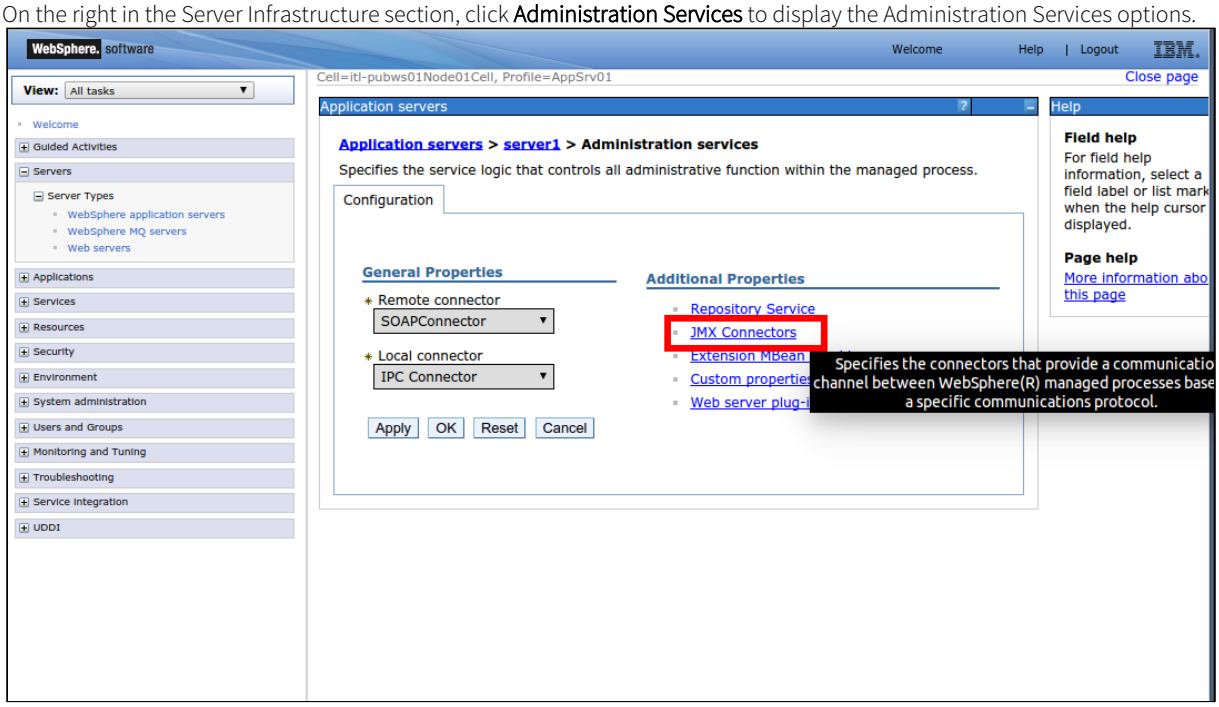
2. Click **Servers** on the left to expand the Servers options.
3. Click **Server Types** to display the Server Types options.
4. Click the **WebSphere Application Servers** link to display the application servers on the right.



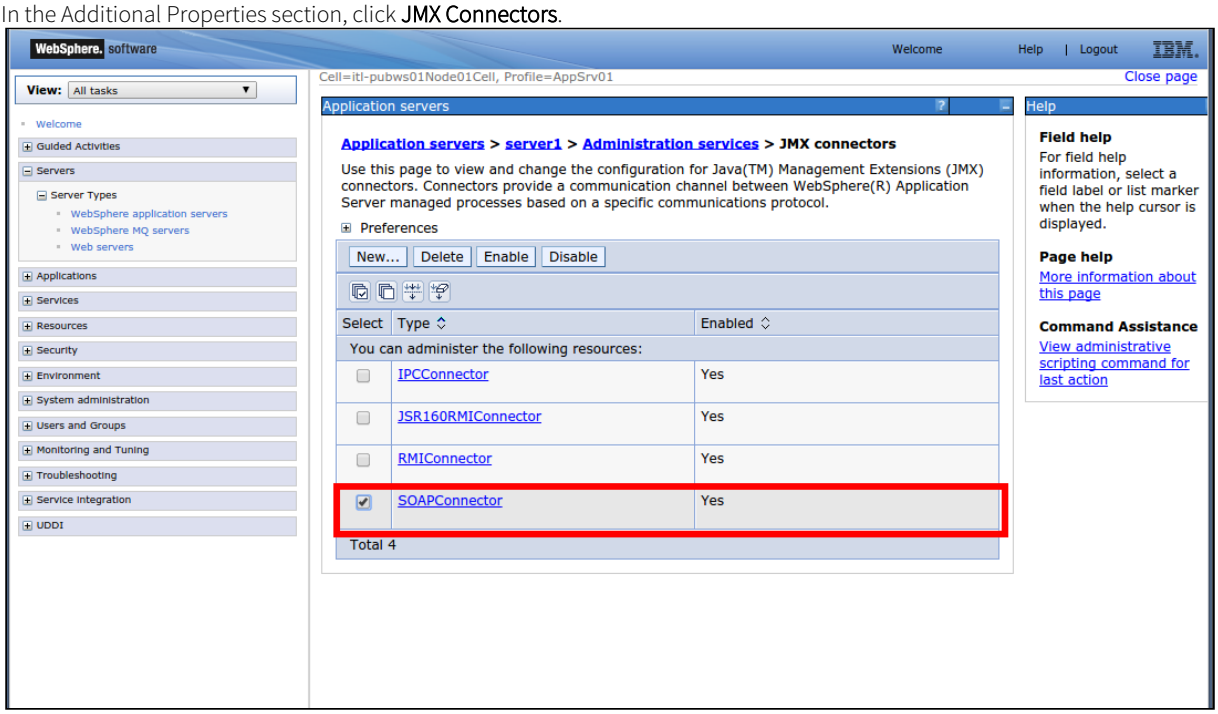
5. Click on the server to administer.



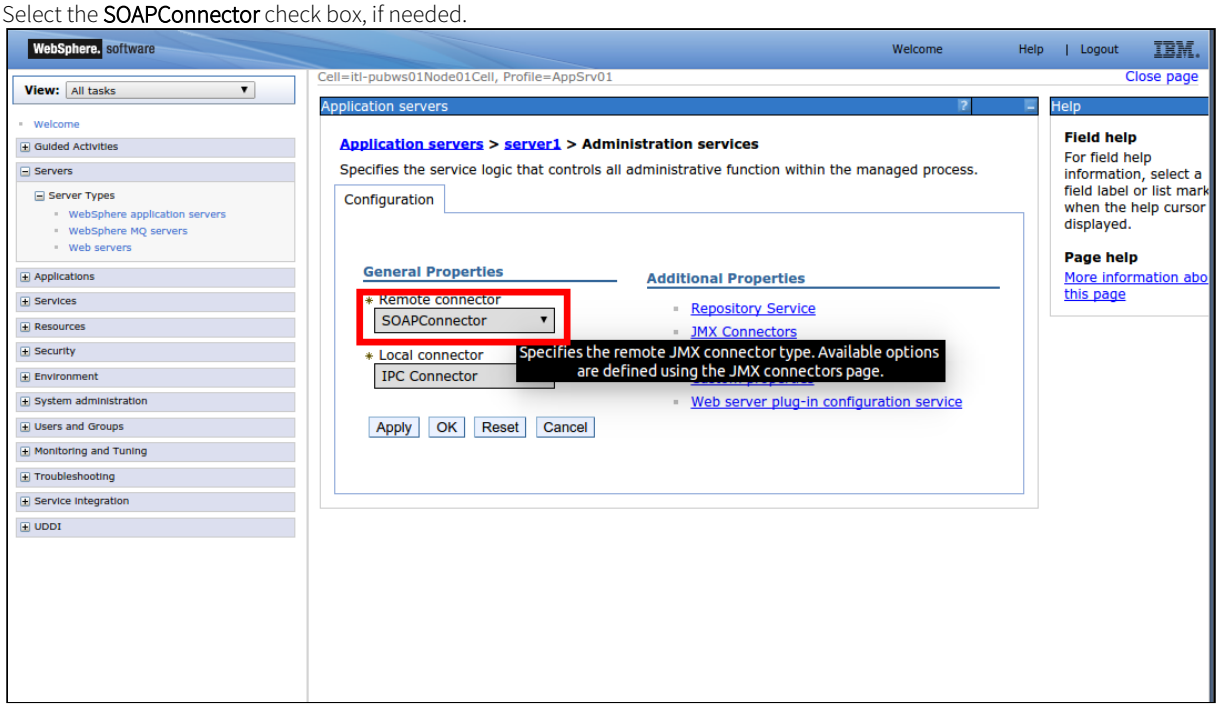
6. On the right in the Server Infrastructure section, click Administration Services to display the Administration Services options.



7. In the Additional Properties section, click **JMX Connectors**.

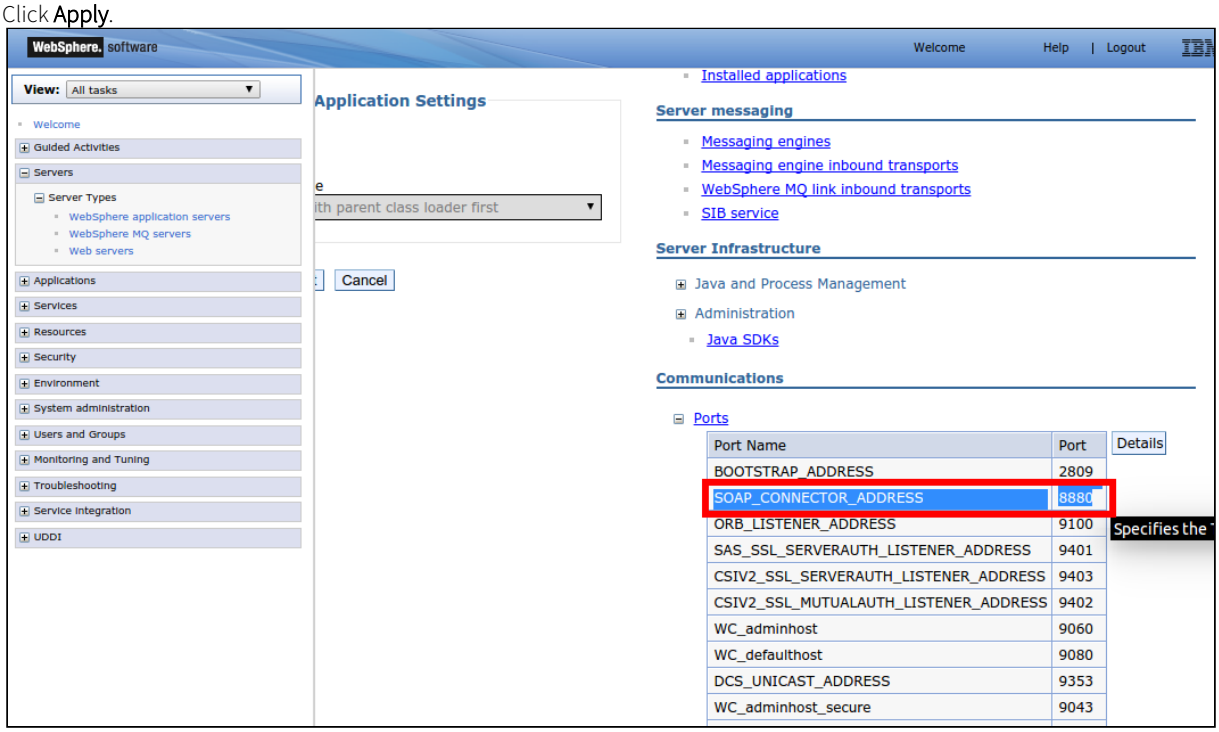


8. Select the **SOAPConnector** check box, if needed.



9. Return to the previous page.
10. In the General Properties section, click the **Remote Connector** drop-down and select **SOAPConnector**.

11. Click **Apply**.



12. Return to the Servers page to find the port for SevOne NMS to use to monitor the device.
13. In the Communications section, click **Ports** to expand the Ports list.
14. Make a note of the **SOAP_CONNECTOR_ADDRESS** port number. This port number is used for the device on the Edit Device page.

25 Enable NBAR

25.1 Cisco NBAR

Network Based Application Recognition (NBAR) is the mechanism some Cisco routers and switches use to inspect sent packets to recognize a dataflow. NBAR is useful for Quality of Service and security purposes such as dealing with malicious software that uses known ports to fake being priority traffic and detecting non-standard applications that use dynamic ports.

This topic describes how to enable NBAR devices to send NBAR data to SevOne NMS. This workflow is outside of the SevOne NMS application and may not present all of the steps your network requires to enable devices to send NBAR data. If the following instructions are not applicable for your network please reference the official Cisco documentation.

Related SevOne NMS workflows include the following.

- The Device Manager provides access to the New Device page and the Edit Device page where you enable the NBAR plugin for a device.
- The NBAR Reports page enables you to run NBAR reports.

25.2 Send NBAR Data to SevOne NMS

Perform the following steps on each interface to enable a Cisco IOS router to send NBAR data and to verify the operation.

1. Enter the following command to enable NBAR.

```
router(conf-if)# ip nbar protocol-discovery
```

2. Enter the following command to verify the operation.

```
router(conf-if)# show ip nbar protocol-discovery
```

26 Enable SNMP

Simple Network Management Protocol (SNMP) is a key technology to manage networks of any size. Virtually all operating systems such as Cisco routers, Linux servers, Extreme switches, and Windows desktop support SNMP. Devices that support SNMP run an SNMP agent that is usually built into the operating system to store information about the device in a tree-like structure. For additional details, see the SNMP topic.

This topic describes how to enable SNMP devices to send SNMP data to SevOne NMS. This workflow is outside of the SevOne NMS application and may not present all of the steps your network requires to enable devices to send SNMP data. If the following instructions are not applicable for your network please reference the device manufacturer's documentation.

Related SevOne NMS workflows include the following.

- The Device Manager provides access to the New Device page and the Edit Device page where you enable the SNMP plugin for a device.
- The Object Types page enables you to view details for the SNMP object types and indicator types the SNMP plugin polls in your network.
- The Object Subtype Manager enables you to view details for the SNMP object subtypes the SNMP plugin polls in your network.
- The MIB Manager enables you to add and manage the MIBs from which you select the OIDs.
- The SNMP OID Browser enables you to select the OIDs to define SNMP object types and SNMP trap events.
- The Indicator Type Map page enables you to enable or disable the device-specific indicators you want the SNMP plugin to poll.

26.1 Send SNMP Data to SevOne NMS

Enter this command to enable SNMP on a Cisco router.

```
router(config)# snmp-server community <YourReadCommunityStringHere> ro
router(config)# snmp-server community <YourWriteCommunityStringHere> rw
router(config)# snmp-server location <Your location here>
router(config)# snmp-server contact <yourAdmin@yourServer.com>
```

Enter this command to verify the operation.

```
router(config)# show snmp
```

26.1.1 SNMP Versions

SevOne NMS uses the following version-specific SNMP commands during device discovery.

Operation	v1	v2c	v3
GET SevOne NMS queries a network device for a single parameter.	`SNMP_MSG_GET`		
SET SevOne NMS queries a parameter and all of its conceptual children from a device.	`SNMP_MSG_SET`		
WALK	`SNMP_MSG_GETNEXT`	`SNMP_MSG_GETBULK`	`SNMP_MSG_GETBULK`

Operation	v1	v2c	v3
Non-repeaters	-	0	
Max-repetitions	-	20	
Timeout Default timeout in seconds for all operations after which SevOne NMS is to give up on a request.	3		
Retries Number of times SevOne NMS retries a request.	2		

SNMP MIBs and OIDs

Each entry in the SNMP structure is a series of numbers, such as .1.3.6.1.2.1.1.1. As you read the number, each number to the right is more specific than the number to its left. Each number to the right is thought of as a child of the number to its left. A string of such numbers is known as an Object Identifier (OID) because it defines a unique identifier for a particular thing, or object.

Management Information Bases (MIBs) define textual names for OIDs. Manufacturers tend to have their own MIBs to describe specific things about their systems.

> **Example**
>
> The name of the OID .1.3.5.1.2.1.1.1 is sysDescr. This OID is used for the system description of a device.

The textual name of an OID is only unique in the MIB to which it belongs so OIDs are accurately written as follows:

```
<MIB Name>::<OID name>
```

26.1.2 Conceptualize Objects, SNMP Walks

The following is a sample SNMP walk of the RFC1213 MIB for a particular device to illustrate what a sample SNMP walk may look like.

```
RFC1213-MIB::ifIndex.1 = INTEGER: 1
RFC1213-MIB::ifIndex.2 = INTEGER: 2
RFC1213-MIB::ifIndex.8 = INTEGER: 8
RFC1213-MIB::ifDescr.1 = STRING: "Ethernet3/0"
RFC1213-MIB::ifDescr.2 = STRING: "Serial3/0"
RFC1213-MIB::ifDescr.8 = STRING: "Loopback0"
RFC1213-MIB::ifType.1 = INTEGER: ethernet-csmacd(6)
RFC1213-MIB::ifType.2 = INTEGER: frame-relay(32)
RFC1213-MIB::ifType.8 = INTEGER: softwareLoopback(24)
RFC1213-MIB::ifSpeed.1 = Gauge32: 10000000
RFC1213-MIB::ifSpeed.2 = Gauge32: 1544000
RFC1213-MIB::ifSpeed.8 = Gauge32: 4294967295
RFC1213-MIB::ifPhysAddress.1 = Hex-STRING: 00 30 80 F3 1F F1
RFC1213-MIB::ifPhysAddress.2 = ""
RFC1213-MIB::ifPhysAddress.8 = ""
RFC1213-MIB::ifAdminStatus.1 = INTEGER: up(1)
RFC1213-MIB::ifAdminStatus.2 = INTEGER: up(1)
RFC1213-MIB::ifAdminStatus.8 = INTEGER: up(1)
RFC1213-MIB::ifOperStatus.1 = INTEGER: up(1)
RFC1213-MIB::ifOperStatus.2 = INTEGER: down(2)
```

```
RFC1213-MIB::ifOperStatus.8 = INTEGER: up(1)
RFC1213-MIB::ifInOctets.1 = Counter32: 1890978658
RFC1213-MIB::ifInOctets.2 = Counter32: 0
RFC1213-MIB::ifInOctets.8 = Counter32: 0
RFC1213-MIB::ifInDiscards.1 = Counter32: 85
RFC1213-MIB::ifInDiscards.2 = Counter32: 0
RFC1213-MIB::ifInDiscards.8 = Counter32: 0
RFC1213-MIB::ifOutOctets.1 = Counter32: 2071381140
RFC1213-MIB::ifOutOctets.2 = Counter32: 0
RFC1213-MIB::ifOutOctets.8 = Counter32: 2819292
RFC1213-MIB::ifOutDiscards.1 = Counter32: 0
RFC1213-MIB::ifOutDiscards.2 = Counter32: 0
RFC1213-MIB::ifOutDiscards.8 = Counter32: 0
```

All of these OIDs refer to three interfaces, identified by the final number of each OID; in this case 1 (for "Ethernet3/0"), 2 (for "Serial3/0"), and 8 (for "Loopback0"). The information about each particular interface is interleaved with that of the others.

Perform the following steps to find out whether Ethernet3/0 is up or down.

1. Search every `ifDescr` entry until you find the one whose value matches Ethernet3/0.
2. Make a note of the index number (the last number) for the Ethernet3/0 ifDescr entry.
3. Check the value for the `ifOperStatus` that uses that index.

26.1.3 SNMP Object

SevOne NMS groups OIDs under the guise of an object. An object is defined by an index value (1, 2, or 8 in the previous example), and may have multiple OIDs which each use the object's index to resolve their values. Each of the OIDs under an object is known as an indicator.

To use the previous example, SevOne NMS conceptually creates an object outlined as below.

```
Object
    Index: 1
        Indicators
            RFC1213-MIB::ifIndex
            RFC1213-MIB::ifDescr
            RFC1213-MIB::ifType
            RFC1213-MIB::ifSpeed
            RFC1213-MIB::ifPhysAddress
            RFC1213-MIB::ifAdminStatus
            RFC1213-MIB::ifOperStatus
            RFC1213-MIB::ifInOctets
            RFC1213-MIB::ifInDiscards
            RFC1213-MIB::ifOutOctets
            RFC1213-MIB::ifOutDiscards
```

Each interface object typically has the same definition but a different index value, therefore, all interfaces to the system are closely related. SevOne NMS monitors every SNMP item that is part of an object.

26.2 Troubleshoot Common SNMP Problems

SNMP version 2 is in common use and includes 64-bit counters and newer MIBs. SNMP version 3 is newer but is not widely supported and you should use SNMP version 1 only when necessary.

26.2.1 Cannot See Interfaces After SNMP Discovery

You may not see an interface because SevOne NMS cannot SNMP walk the device due to an incorrect community string. Perform the following steps to verify the SNMP community strings for the device are correct.

1. Go to the Device Manager.
2. Click Next to the device to display the Edit Device page.
3. On the SNMP plugin, verify the community string and SNMP version are correct.
4. If they are correct, log on to the device and enter this command to walk the device: *snmpwalk -v<version> -c<community string> <ip address>* (If the SNMP version is 2, use "2c" for the version.)

5. If the command fails, then SevOne NMS cannot SNMP walk the device. Try to walk the device from another location to ensure that the device is properly configured.

Common reasons for not being able to SNMP walk a device include:

- Routing - There is no route from SevOne NMS to the device.
- Firewall - A router between SevOne NMS and the device blocks SNMP traffic.

26.2.2 Interface Synchronization Settings

The Cluster Manager > **Cluster Settings** tab enables you to synchronize the object state in SevOne NMS with the operational and/or administrative enable/disable (up/down) state on the actual device.

The SNMP plugin on the Edit Device page enables you to select from the following options to synchronize the interface administrative status and to synchronize the interface operational status:

- Select **Auto** to use the Cluster Manager > **Cluster Settings** tab setting for object synchronization (administrative state and operational state).
- Select **On** to override the Cluster Manager to disable and hide the administratively/operationally down interfaces and to enable and show the administratively/operationally up interfaces.
- Select **Off** to override the Cluster Manager to take no action in regards to enabling or disabling objects based on their administrative/operational status.

26.2.3 Alert Not Received When Event Occurs

SevOne NMS collects all traps and provides a collection of common trap events. Traps that have a corresponding trap event appear on the Logged Traps page and traps that do not have a corresponding trap event appear on the Unknown Traps page. The Unknown Traps page provides a Configure Trap Event button to provide access to the Trap Event Editor where you configure traps for your network.

If you think you should receive an alert on a trap from a device, go to the Unknown Traps page and use the filters to search by the IP address of the device.

If a trap does not resolve to an OID name, then the OID for the trap is not included in SevOne NMS. The trap is still processed and appears as an OID number instead of a name.

27 SNMP

27.1 SNMP As Seen By SevOne NMS

SevOne SNMP Scripting (S3) is a scripting language developed by SevOne to enable the SNMP plugin to discover complex SNMP objects. The primary purpose of S3 is to describe an SNMP OID in relation to other SNMP OIDs. S3 creates text that relates to and is based on OIDs in order to create more user friendly SNMP object names and descriptions. S3 also relates OIDs to each other via logical and mathematical expressions to store the resultant value for later use such as to cross reference OIDs across a device SNMP tree and to gather descriptive information about a particular object.

The SNMP plugin uses S3 for the following:

- Define an object name.
- Define an object description.
- Define an object type.
- Determine which objects to include.
- Define an indicator.

27.2 SNMP Object Naming Process

The SNMP plugin uses the following scoring formula to name a newly discovered object.

Assume that the best score to begin is 0.

1. Go through all of the objects for the device.
2. If the existing object's object type is the wrong name, skip.
    - The best possible score is 5.
3. Assume that the new object description is valid.
    - If the new object description is the same as the Object Type name, then the object description is no good.
    - If two things that the SNMP plugin already found have the object description, then the object description is no good.
4. Assume that the old description is valid.
    - If the existing object description is the same as the Object Type name, then the object description is no good.
    - If two existing objects already have that object description, then the object description is no good.

    If either object description is no good, then the best possible score is now 4.
    The current score is 0.
5. If the object names are the same, then increase the current score by 3.
    - Otherwise, if the new object description is good and the existing object name is the same as the new object description, then increase the current score by 1.
6. If both object descriptions are good and the same, then increase the score by 1.
    - Otherwise, if the existing object description is good and the existing object description is the same as the new object name, then increase the current score by 1.
7. If the SNMP indexes are the same, then increase the current score by 1.
8. If the score is at least 2, and the score is better than the best score so far, consider the objects the same.

Human breakdown:

```
/ / Scoring analysis.
 //
 // Possible scores: / Current name matches existing name. [+3]
 // 3,1,0            | Current description is good.
 //                  \    Current description matches existing name. [+1]
 //
 // Possible scores: / Existing description is good.
 // 1,0              |    Current description is good.
 //                  |       Existing description matches current
description. [+1]
 //                  \    Existing description matches current name. [+1]
 //
 // Possible scores: / Current index matches existing index. [+1]
 // 1,0              \
 //
```

```
// So, when are things the same?
// 1. The names are the same.
// 2. The description (if good) matches an existing name AND the
descriptions (if good) are the same.
// 3. The description (if good) matches an existing name AND the name is
matches an existing description (if good).
// 4. The description (if good) matches an existing name AND the indexes are
the same.
// 5. The name is matches an existing description (if good) AND the indexes
are the same.
```

So what does this mean?

SevOne NMS discovers and polls a router with two network cards with two ports on each.

All objects belong to the Interfaces object type.

Each port is an object.

Object Name	Object Description	SNMP Index	Note	Score
Eth0	Internet Access	1	Existing object with poll data	n/a
Eth1	Interface	2	Existing object with poll data	n/a
Eth2	Site One	3	Existing object with poll data	n/a
Eth3	Back Link	4	Existing object with poll data	n/a

You rename the ports on the router. Upon rediscovery the SNMP plugin continues to poll the objects with no change or data loss as long as you do not change the object description or the SNMP index.

Object Name	Object Description	SNMP Index	Note	Score
Ethernet0	Internet Access	1	No data lost, polled as if it were still Eth0	x
Ethernet1	Interface	2	No data lost, polled as if it were still Eth1	x
Ethernet2	Site One	3	No data lost, polled as if it were still Eth2	x
Ethernet3	Back Link	4	No data lost, polled as if it were still Eth3	x

You remove the first card from the router. The router automatically renames each port. Upon rediscovery the SNMP plugin continues to polls the objects with no change or data loss as long as you do not change the description or the index.

Object Name	Object Description	SNMP Index	Note	Score
~~*Ethernet0*~~	~~*Internet Access*~~	~~*1*~~	Data stored for the number of Days Until Delete setting in the Cluster Manager	x
~~*Ethernet1*~~	~~*Interface*~~	~~*2*~~	Data stored for the number of Days Until Delete setting in the Cluster Manager	x
Ethernet0	Site One	3	No data lost, polled as if it were still Ethernet2	x

Ethernet1	Back Link	4	No data lost, polled as if it were still Ethernet3	x

You add the card back to the router within the number of Days Until Delete setting in the Cluster Manager. The router automatically changes the object name.

Object Name	Object Description	SNMP Index	Note	Score
Ethernet0	Internet Access	1	Data gap for when the card was removed but otherwise no data lost, polled as if it were still Ethernet0	x
Ethernet1	Interface	2	Data gap for when the card was removed but otherwise no data lost, polled as if it were still Ethernet1	x
Ethernet2	Site One	3	No data lost, polled as if it were still Ethernet0 from previous discovery.	x
Ethernet3	Back Link	4	No data lost, polled as if it were still Ethernet1 from previous discovery.	x

You decide to rename port four and change its description.

Object Name	Object Description	SNMP Index	Note	Score
Ethernet0	Internet Access	1	No data lost, polled as if it were still Ethernet0 from previous discovery.	x
Ethernet1	Interface	2	No data lost, polled as if it were still Ethernet1 from previous discovery.	x
Ethernet2	Site One	3	No data lost, polled as if it were still Ethernet2 from previous discovery.	x
Eth3	**Site Three**	4	**New object created and new poll data collected.** Existing data stored for the number of Days Until Deleted setting in the Cluster Manager.	x

27.3 Anatomy of SNMP Data

At an extraordinarily high level, SevOne NMS groups each SNMP object by an SNMP object type. Each object has indicators that are grouped by indicator type. SNMP Objects are composed of OIDs which in turn break down into MIBs.

27.3.1 How OIDs Are Indexed

Device manufacturers name SNMP OIDs by a series of numbers, (e.g., The OID sysDescr is .1.3.6.1.2.1.1.1). Everything that comes after a named OID is the OID index. System wide information is usually denoted as .0 information. The .0 generally means there is only one instance of the OID, (e.g., The sysDescr index is always .0). The OID ifDescr is indexed by a single number that is not 0, (e.g., The description of an Ethernet card in a server might be ifDescr.3). There is no limit to the count of numbers that follow an OID. Some manufacturers use integer indexes that have a constant amount of numbers that follow each OID and some manufacturers use string

indexes for readability. A string index represents a piece of text (the string) as a series of characters where each character is represented as its ASCII value. String indexes are sometimes prefixed with the length of the string.

27.3.1.1 Index Types

Integer - A single number of any size.

> A simple integer that could be used for an interface index.
> `7`

String (with length) - A string of text, prefixed with the number of characters and followed by the ASCII value of each character.

> The text CPU is the number of characters followed by the ASCII value of each.
> `3.67.80.85`

String (no length) - A string where the length is not prefixed.

> The text CPU in ASCII values.
> `67.80.85`

Number (with length) - A string index where the numbers do not have an ASCII meaning.

> The IP address with its length before the octets.
> `4.192.158.0.1`

Number (no length) - A string index with no length where the numbers do not have an ASCII meaning.

> The IP address with no length information appears as a series of integer indexes. This is useful when there is no guaranteed how many numbers there could be.
> `192.168.0.1`

27.4 How S3 Handles SNMP

S3 uses the full numeric OID representation (starting with .1) to remove the dependency on the MIBs and to remove potential ambiguity of the OID. This makes the object definition fully portable to another system because different MIBs can arbitrarily redefine OIDs.

Another S3 feature is that white space characters such as spaces, tabs, and new line characters are optional and exist for readability. All of the following are equivalent:

- 7+9
- 7 + 9
- 7
+9
- 7
+
9

27.4.1 Scripts

An S3 script is a sequential evaluation of one or many statements. Each statement is executed in sequence. The only logic provided by S3 comes from flavors of the ternary operator which acts like an IF statement. The final result of the script is the result of the final statement in the script.

Example 1:

```
Statement 1
```

Example 2:

1. `Statement 1`
2. `Statement 2`
3. `Statement 3`

27.4.1.1 Statements

A statement is the atomic unit of a script. A statement can assign a value to a variable or a statement can be an expression that evaluates to some value. Each statement evaluates to some value upon execution, (e.g., For a variable assignment, the value of the statement is the value of the variable).

Statements are lists of expressions and expression chains may be very complex and long. All S3 statements must end in a semicolon <;>. The last statement in a script may omit the semicolon.

Simple statement:

```
Expression ;
```

List statement where the final value is the concatenation of both expressions:

```
Expression 1 Expression 2 ;
```

27.4.1.2 Expressions

An expression is the atomic unit of a statement. S3 is an OID-evaluation language and a text creation language. Multiple expressions can lie next to each other and their results are concatenated together. This differs from more logical and mathematical languages.

Example: `1 + 2 3 + 4`

- In C there needs to be a joining operation between the 2 and the 3 because they are considered two disjointed expressions which results in a syntax error.
- In S3 this is seen as two separate expressions next to each other: 1+ 2 and 3 + 4 and the result of the statement is 37. The white space does not matter unless enclosed in quotes.

An expression is anything that evaluates to a value. This value need not be numeric. A piece of text evaluates to itself. An expression might be the number 7, the word Hello, or a complex mathematical formula. Expressions can be chains of symbols and operators as long as the entire expression evaluates to a single value.

- Number

  ```
  7
  ```

- String:

  ```
  'Hello'
  ```

- Complex Formula:

  ```
  ( ( 1 + 2 ) / 12 + 34 ) * 10
  ```

- Variable or OID that evaluates to a number or string:

  ```
  [INDEX]
  .1.3.6.1.2.1.1.1.0
  ```

- Multiple grouped expressions (enclosed within parentheses) concatenated together:

  ```
  ( 7 'Hello' ( ( 1 + 2 ) / 12 + 34 ) * 10 [INDEX] .1.3.6.1.2.1.1.1.0 )
  ```

27.4.1.3 Variables

Variables are evaluated as OIDs to store the value of an expression. S3 has two types of variables; scalars and vectors.

- Scalar - Anything that is a single number or some text.
- Vector - An array of things. Vectors in S3 are different from vectors in normal scripting languages. Vectors in S3 are geared toward OIDs because an individual OID is represented as .<number> and a full OID is a series of .<numbers>s one after another. S3 breaks down variables into vectors by the "." character.

Variables are a name surrounded by square brackets. Variable names consist of the following characters: a-z, A-Z, 0-9, and - .

```
[Variable Name]
```

A variable assignment is an expression. The evaluation of the assignment is the new variable value. A variable assignment uses the = operator.

```
[Variable name] = Expression
```

S3 uses the following conventions to differentiate SevOne system variables from user variables to prevent user variables from overwriting system variables. There is no rule to enforce this.

- SevOne system variables use capital letters and underscores for spaces. `[MY_VARIABLE]`
- User variables use lowercase letters, a capital letter in the next word, and no underscore for spaces. `[myVariable]`

S3 can treat and evaluate variables as OIDs. Each variable must be declared before use. There is no special declaration syntax, but a variable must have an assigned value before use in an expression. Both scalar variables and vector variables are evaluated and inserted raw into the expression. S3 does nothing special to scalar variables when scalar variables are evaluated.

When a vector variable is evaluated, each of the vector variable's components is written, separated by "."s. Elements in vector variables are zero-indexed numerically. The first element starts at 0, the second starts at 1, and so on. To access a particular element of a vector variable, surround the element index in curly braces after the variable.

```
[Variable Name ]{Index number }
```

A variable cannot be used as an index number. The index number must be an actual number.

Each element in a vector variable is usually a scalar variable. There are exceptions when an element in a vector variable is another vector variable.

Some variables should not be evaluated as an OID. Enclose the variable in back ticks to prevent the variable from being evaluated as an OID. If a variable simply contains a number, the variable is treated as a normal number if not back ticked; however, it is always safe to back tick a variable to prevent improper evaluation. Text evaluates to itself. Text is considered anything enclosed in quotes. Back ticks may be in a single quoted string and that single quote string may be in a back tick quoted string.

- Single quotes (') - Single quotes are used for raw text. The content of the text is not processed in any way, e.g., `'Anything here'`
- Back Ticks (`) - Back ticks are used for variable interpolation. Any variables present in the text is evaluated, e.g., `` `Anything here, including variables` ``

27.4.2 S3 OID Handling

OIDs are evaluated. Anything that is not text, a variable, an operator, a normal number, or otherwise special symbol is considered an OID. When an OID is evaluated, it evaluates to the value of the OID on the current device. If the OID is not present on the device, the OID, followed by the default SNMP index, is used instead. If the OID cannot be evaluated, it evaluates to the empty string.

It is critical to note here that a variable without back ticks around it is treated exactly as it would if the value of the variable were to be placed in its stead. This means that a variable that contains a string representation of an OID is evaluated as that OID when it is encountered.

The following example might not work as expected:

```
1 [test] = 'test';
2 [test1] = [test] 1;
```

One might expect the value of "[test1]" to be "test1"; however, since "[test]" is not back ticked, it is treated as if the text "test" were present. As such, S3 tries to get the value of the OID whose name is "test" naturally, there is not one, and it returns the empty string. Thus, the final value is actually "1".

The proper way to do this is:

```
1 [test] = 'test';
2 [test1] = `[test]` 1;
```

This example back ticks the variable to prevent it from being evaluated as an OID.

27.4.2.1 Indexing

When an OID is encountered, S3 tries to evaluate it. If S3 cannot evaluate the OID, then S3 adds the value of the default index to the OID, which for SNMP discovery is [INDEX]. If S3 still cannot evaluate the OID, then the OID evaluates to the empty string. This allows

for very human readable and human understandable definitions for objects and indicators. However, at the loss of stringent definitions.

If an OID already has .[INDEX] appended to it, then the OID saves S3 the step.

27.4.3 Symbols

Symbols are special tokens (characters, or collections of characters) that have a special function in S3.

27.4.3.1 Grouping

Parentheses (and) group expressions to define the sequence in which they are to be evaluated. This is commonly used in mathematical applications.

Example:

```
1 + 2 * 3
```

(which is 7)

Is not the same as:

```
( 1 + 2 ) * 3
```

(which is 9).

Parentheses can change two expressions into one expression.

Example:

```
( 1 + 2 3 + 4 )
```

Evaluates to the single value 37, which could be used by further expressions.

27.4.3.2 Operators

Operators are symbols. Operators are anything that act on an expression. There are three types of operators:

- Unary operators act on one value only, (e.g., Not).
- Binary operators act on two values, (e.g., Addition).
- Ternary operators act on three values, (e.g., ... ? ... : ... is a ternary operator in C).

27.4.3.3 Math

The common mathematical operators are applied with the usual precedence. Mathematical operators have full floating point support.

Multiplication (Standard multiplication)

```
Left expression * Right expression
```

Division (Standard division)

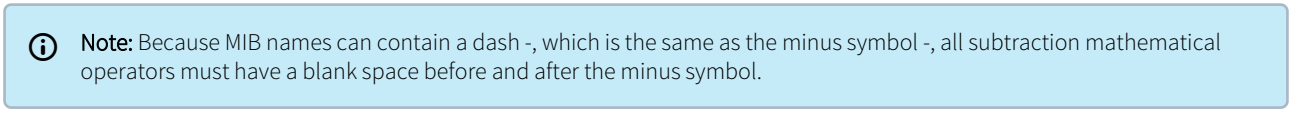
```
Left expression / Right expression
```

Addition (Standard addition)

```
Left expression + Right expression
```

Subtraction (Standard subtraction)

```
Left expression - Right expression
```



27.4.3.4 Comparison

Comparison operators compare two expressions that return 1 if the comparison is true or return 0 if the comparison is false.

Equal to, Boolean == returns 1 if the left and right side are equal.

```
Left expression == Right expression
```

Not equal to, Boolean != returns 1 if the left and right side are not equal.

```
Left expression != Right expression
```

Less than, Boolean < returns 1 if the left side is less than the right side.

```
Left expression < Right expression
```

Less than or equal to, Boolean <= returns 1 if the left side is less than or equal to the right side.

```
Left expression <= Right expression
```

Greater than, Boolean > returns 1 if the left side is greater than the right side.

```
Left expression > Right expression
```

Greater than or equal to, Boolean >= returns 1 if the left side is greater than or equal to the right side.

```
Left expression >= Right expression
```

27.4.3.5 Logic

Logical operators generally perform true/false operations. S3 uses the following logical operators:

Binary

Binary logical operators operate on two expressions.

Logical AND, Boolean && returns 1 if the left and right side evaluate to true or returns 0 otherwise.

```
Left expression && Right expression
```

Logical OR, Boolean || returns 1 if the left side evaluates to true, the right side evaluates to true, or both evaluate to true; or returns 0 otherwise.

```
Left expression || Right expression
```

Bamboo, ||| is actually a shortcut for a particularly common case of the ?? ternary operator. It returns the value on the left if it is set; otherwise, it returns the value on the right regardless of its value.

```
Left expression ||| Right expression
```

This is equivalent to

```
Left expression ?? Left expression : Right expression
```

27.4.3.6 Ternary

Ternary logical operators operate on three expressions and S3 has two ternary operators.

Logical ternary operator, ? evaluates the left expression for a test that is greater than 0 (numerically) or for a string that has length and is not 0. Otherwise, it evaluates the right expression. For this reason, the test is usually a logical Boolean expression that returns 0 or 1, guaranteed.

```
test ? Left expression : Right expression
```

Existential ternary operator, ?? evaluates the left expression for a test that has a value that is not the empty string. Otherwise, it evaluates the right expression.

```
test ?? Left expression : Right expression
```

Note:

```
test ?? test : Right expression
```

Is equivalent to:

```
test ||| Right expression
```

27.4.3.7 Count

The results of a walk, #count walks the specified OID and returns the count of the occurrences of an OID. This does not resolve the OID in the manner that other naked OIDs are resolved to get the OID value. This #count resolves the OID count immediately, unlike the way an OID is resolved via an OID walk.

Example: To count the number of CPUs on a Linux device to determine what the maximum CPU utilization could be: net-snmp returns up to 800% utilization for a box with eight CPUs.

```
#count OID
```

Example:

```
#count .1.3.6.1.2.1.25.3.3.1.2
```

Can evaluate to 8.

27.4.3.8 Conversion

Since OIDs may be indexed by numbers, strings, or variably-sized components, S3 uses conversion operators that operate on a single expression.

Conversion from OID

OID-to-ASCII-string (with length), $s converts the expression to an ASCII string.

```
$s Expression
```

The expression should be an OID with the following format:

```
n.ASCII 1.ASCII 2.....ASCII n
```

Example:

```
$s '5.72.101.108.108.111'
```

Evaluates to Hello.

OID-to-ASCII-string (no length), $S converts the expression to an ASCII string.

```
$S Expression
```

The expression should be an OID with the following format:

```
ASCII 1.ASCII 2
```

Example:

```
$S '72.101.108.108.111'
```

Evaluates to Hello.

OID-to-numbers (with length), $v converts the expression to a string.

```
$v Expression
```

The expression should be an OID with the following format:

```
n.Number 1.Number 2.....Number n
```

Example:

```
$v '4.192.168.0.1'
```

Evaluates to 192.168.0.1.

OID-to-numbers (no length), $V Identity operation; the value should be the same as the expression.

```
$V Expression
```

This converts the expression to a string. The expression should be an OID with the following format:

```
Number 1.Number 2.
```

Example:

```
$V '192.168.0.1'
```

Evaluates to 192.168.0.1.

27.4.3.9 Conversion to OID

String-to-OID (with length), #s converts the expression to an OID with the length prefixed. The expression should be ASCII text.

```
#s Expression
```

Example:

```
#s 'Hello'
```

Evaluates to 5.72.101.108.108.111.

String-to-OID (no length), $s converts the expression to a string with no length information. The expression should be ASCII text.

```
#S Expression
```

Example:

```
#S 'Hello'
```

Evaluates to 72.101.108.108.111.

Numbers-to-OID (with length), #v converts the expression to an OID with the length prefixed. The expression should be text consisting of numbers separated by "."s.

```
#v Expression
```

Example:

```
#v '192.168.0.1'
```

Evaluates to 4.192.168.0.1.

Numbers-to-OID (no length), #V Identity operation; the value should be the same as the expression converts the expression to an OID with no length information. The expression should be text consisting of numbers separated by "."s.

```
#V Expression
```

Example:

```
#V '192.168.0.1'
```

Evaluates to 192.168.0.1.

27.4.3.10 Grouping

Parameters to the conversion operators should be enclosed in parentheses to avoid confusion.

To get the OID index representation of the text 37 (which is 2.51.55), you can try:

```
#s 1 + 2 3 + 4
```

However, the #s only applies to the 1 which yields 1.49. (49 is ASCII for 1); the value of that is added to 2 (1.49 + 2 = 3.49), which is then concatenated with 7 (to yield 3.497). You must use parentheses:

```
#s ( 1 + 2 3 + 4 )
```

Evaluates to 2.51.55 (which, as a string, is 37).

27.4.4 Precedence

The precedence of operators and symbols is as follows. When given the choice (in other words, when parentheses are not used), S3 evaluates operations in the following sequence.
The normal mathematical operator precedence (* / + -) is preserved in this list.

1. 'text' `text`
2. ()
3. #s #S #v #V $s $S $v $V
4. * /
5. + -
6. == != > >= < <=
7. &&
8. ||
9. |||
10. :
11. ? ??
12. =

27.4.5 Variable Assignment Example

The following examples assign the proper values to variables whose name should match their value.

1. [one] = 1;
2. [two] = 1 + 1;
3. [two] = `[one]` + `[one]` - 1 + 1;
4. [ten] = (2 + 1) * 3 + 1;

The following example sets all three variables equal to 12.

```
1 [x] = [y] = [z] = 12;
```

27.4.5.1 Logic

The following examples demonstrate Boolean logic.

1. [bothXandY] = `[x]` && `[y]`;
2. [eitherXorYorBoth] = `[x]` || `[y]`;
3. [eitherXorY] = (`[x]` && (`[y]` ? 0 : 1)) || (`[y]` && (`[x]` ? 0 : 1));
4. [notX] = `[x]` ? 0 : 1;

The following example selects the value of `ifName` if it is present, or the value of `ifDescr` otherwise.

```
[bamboo] = ifName ||| ifDescr;
```

The following examples demonstrate the use of the ternary operator.

1. [sevenOrEight1] = `[x]` ? 7 : 8;
2. [sevenOrEight2] = `[x]` ? 6 + 1 : 2 * 2 + 4;

27.4.5.2 Conversion

The following examples convert the text CPU into an OID index. The ASCII value for C=67, P=80, and U=85.

```
#s 'CPU'
```

The result of this is "3.67.80.85".

```
#S 'CPU'
```

The result of this is 67.80.85 (no length prefix).

The following examples convert the OID indexes specified into strings.

```
$s '3.67.80.85'
```

The result of this is CPU.

```
$S '67.80.85'
```

The result of this is also CPU.

27.4.6 Full Examples

27.4.6.1 VACM Entry

The Net-SNMP agent supports the NET-SNMP-VACM-MIB, which provides some information about Net-SNMP's View Access Control Model.

The following is a sample walk of `nsVacmAccessEntry (.1.3.6.1.4.1.8072.1.9.1.1)`:

1. NET-SNMP-VACM-MIB::nsVacmContextMatch."grpcomm1".".0.noAuthNoPriv."read"
2. = INTEGER: prefix(2)
3. NET-SNMP-VACM-MIB::nsVacmContextMatch."grpcomm1".".0.noAuthNoPriv."write"
4. = INTEGER: prefix(2)
5. NET-SNMP-VACM-MIB::nsVacmContextMatch."grpcomm1".".0.noAuthNoPriv."notify"
6. = INTEGER: prefix(2)
7. NET-SNMP-VACM-MIB::nsVacmContextMatch."grpsnmpUser".".3.authNoPriv."read"
8. = INTEGER: prefix(2)
9. NET-SNMP-VACM-MIB::nsVacmContextMatch."grpsnmpUser".".3.authNoPriv."write"
10. = INTEGER: prefix(2)
11. NET-SNMP-VACM-MIB::nsVacmContextMatch."grpsnmpUser".".3.authNoPriv."notify"
12. = INTEGER: prefix(2)
13. NET-SNMP-VACM-MIB::nsVacmViewName."grpcomm1".".0.noAuthNoPriv."read"
14. = STRING: *all*
15. NET-SNMP-VACM-MIB::nsVacmViewName."grpcomm1".".0.noAuthNoPriv."write"
16. = STRING: none
17. NET-SNMP-VACM-MIB::nsVacmViewName."grpcomm1".".0.noAuthNoPriv."notify"
18. = STRING: none
19. NET-SNMP-VACM-MIB::nsVacmViewName."grpsnmpUser".".3.authNoPriv."read"
20. = STRING: *all*
21. NET-SNMP-VACM-MIB::nsVacmViewName."grpsnmpUser".".3.authNoPriv."write"
22. = STRING: *all*
23. NET-SNMP-VACM-MIB::nsVacmViewName."grpsnmpUser".".3.authNoPriv."notify"
24. = STRING: *all*
25. NET-SNMP-VACM-MIB::nsVacmStorageType."grpcomm1".".0.noAuthNoPriv."read"
26. = INTEGER: permanent(4)
27. NET-SNMP-VACM-MIB::nsVacmStorageType."grpcomm1".".0.noAuthNoPriv."write"
28. = INTEGER: permanent(4)
29. NET-SNMP-VACM-MIB::nsVacmStorageType."grpcomm1".".0.noAuthNoPriv."notify"
30. = INTEGER: permanent(4)
31. NET-SNMP-VACM-MIB::nsVacmStorageType."grpsnmpUser".".3.authNoPriv."read"
32. = INTEGER: permanent(4)
33. NET-SNMP-VACM-MIB::nsVacmStorageType."grpsnmpUser".".3.authNoPriv."write"
34. = INTEGER: permanent(4)
35. NET-SNMP-VACM-MIB::nsVacmStorageType."grpsnmpUser".".3.authNoPriv."notify"
36. = INTEGER: permanent(4)
37. NET-SNMP-VACM-MIB::nsVacmStatus."grpcomm1".".0.noAuthNoPriv."read"
38. = INTEGER: active(1)
39. NET-SNMP-VACM-MIB::nsVacmStatus."grpcomm1".".0.noAuthNoPriv."write"
40. = INTEGER: active(1)
41. NET-SNMP-VACM-MIB::nsVacmStatus."grpcomm1".".0.noAuthNoPriv."notify"
42. = INTEGER: active(1)
43. NET-SNMP-VACM-MIB::nsVacmStatus."grpsnmpUser".".3.authNoPriv."read"
44. = INTEGER: active(1)
45. NET-SNMP-VACM-MIB::nsVacmStatus."grpsnmpUser".".3.authNoPriv."write"
46. = INTEGER: active(1)
47. NET-SNMP-VACM-MIB::nsVacmStatus."grpsnmpUser".".3.authNoPriv."notify"
48. = INTEGER: active(1)

And with numeric OIDs:

1. .1.3.6.1.4.1.8072.1.9.1.1.2.8.103.114.112.99.111.109.109.49.0.0.1.4.114.101.97.100
2. = INTEGER: prefix(2)
3. .1.3.6.1.4.1.8072.1.9.1.1.2.8.103.114.112.99.111.109.109.49.0.0.1.5.119.114.105.116.101
4. = INTEGER: prefix(2)
5. .1.3.6.1.4.1.8072.1.9.1.1.2.8.103.114.112.99.111.109.109.49.0.0.1.6.110.111.116.105.102.121
6. = INTEGER: prefix(2)
7. .1.3.6.1.4.1.8072.1.9.1.1.2.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.4.114.101.97.100
8. = INTEGER: prefix(2)
9. .1.3.6.1.4.1.8072.1.9.1.1.2.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.5.119.114.105.116.101
10. = INTEGER: prefix(2)
11. .1.3.6.1.4.1.8072.1.9.1.1.2.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.6.110.111.116.105.102.121
12. = INTEGER: prefix(2)
13. .1.3.6.1.4.1.8072.1.9.1.1.3.8.103.114.112.99.111.109.109.49.0.0.1.4.114.101.97.100
14. = STRING: *all*
15. .1.3.6.1.4.1.8072.1.9.1.1.3.8.103.114.112.99.111.109.109.49.0.0.1.5.119.114.105.116.101
16. = STRING: none
17. .1.3.6.1.4.1.8072.1.9.1.1.3.8.103.114.112.99.111.109.109.49.0.0.1.6.110.111.116.105.102.121
18. = STRING: none
19. .1.3.6.1.4.1.8072.1.9.1.1.3.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.4.114.101.97.100
20. = STRING: *all*
21. .1.3.6.1.4.1.8072.1.9.1.1.3.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.5.119.114.105.116.101
22. = STRING: *all*
23. .1.3.6.1.4.1.8072.1.9.1.1.3.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.6.110.111.116.105.102.121
24. = STRING: *all*
25. .1.3.6.1.4.1.8072.1.9.1.1.4.8.103.114.112.99.111.109.109.49.0.0.1.4.114.101.97.100
26. = INTEGER: permanent(4)
27. .1.3.6.1.4.1.8072.1.9.1.1.4.8.103.114.112.99.111.109.109.49.0.0.1.5.119.114.105.116.101
28. = INTEGER: permanent(4)
29. .1.3.6.1.4.1.8072.1.9.1.1.4.8.103.114.112.99.111.109.109.49.0.0.1.6.110.111.116.105.102.121
30. = INTEGER: permanent(4)
31. .1.3.6.1.4.1.8072.1.9.1.1.4.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.4.114.101.97.100
32. = INTEGER: permanent(4)
33. .1.3.6.1.4.1.8072.1.9.1.1.4.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.5.119.114.105.116.101
34. = INTEGER: permanent(4)
35. .1.3.6.1.4.1.8072.1.9.1.1.4.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.6.110.111.116.105.102.121
36. = INTEGER: permanent(4)
37. .1.3.6.1.4.1.8072.1.9.1.1.5.8.103.114.112.99.111.109.109.49.0.0.1.4.114.101.97.100
38. = INTEGER: active(1)
39. .1.3.6.1.4.1.8072.1.9.1.1.5.8.103.114.112.99.111.109.109.49.0.0.1.5.119.114.105.116.101
40. = INTEGER: active(1)
41. .1.3.6.1.4.1.8072.1.9.1.1.5.8.103.114.112.99.111.109.109.49.0.0.1.6.110.111.116.105.102.121
42. = INTEGER: active(1)
43. .1.3.6.1.4.1.8072.1.9.1.1.5.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.4.114.101.97.100
44. = INTEGER: active(1)
45. .1.3.6.1.4.1.8072.1.9.1.1.5.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.5.119.114.105.116.101
46. = INTEGER: active(1)
47. .1.3.6.1.4.1.8072.1.9.1.1.5.11.103.114.112.115.110.109.112.85.115.101.114.0.3.2.6.110.111.116.105.102.121
48. = INTEGER: active(1)

The following entries use the "nsVacmStatus" OID.

27.4.6.2 Indexing

S3 has two options to properly index entries.

1. S3 can choose a variable-length index (with no length prefix). However, this provides S3 no insight as to the components of the index. There are no OIDs available to determine a proper name for any of the entries to enter into the system as objects.
2. S3 can explicitly define each index component, which allows S3 to reference each component individually to properly name objects.

The index composed of the following types of fields.

Example:

```
"grpcomm1"."".0.noAuthNoPriv."read"
```

- String, (e.g., grpcomm1) - According to the MIB, this is the name of the group for this entry.
- String, (e.g., "") - According to the MIB, this is the prefix that a name must match to gain access rights.
- Integer, (e.g., 0) - According to the MIB, this is the security model in use. This roughly corresponds with the SNMP version (where 0 = any).
- Integer, (e.g., noAuthNoPriv, which is, 1) - According to the MIB, this is the minimum level of security required to gain access rights.
- String, (e.g., read) - According to the MIB, this is the type of processing to which to apply the specified view.

S3 has the following information:

- [INDEX] is 8.103.114.112.99.111.109.109.49.0.0.1.4.114.101.97.100.
- [INDEX]{0} is 8.103.114.112.99.111.109.109.49.
- [INDEX]{1} is 0.
- [INDEX]{2} is 0.
- [INDEX]{3} is 1.
- [INDEX]{4} is 4.114.101.97.100.

27.4.6.2.1 Naming

S3 uses the following information to name an object for a VACM entry.

Group *group name* [matching *prefix*] using *{*any version|*security model }*with *security level* , providing *processing*

27.4.6.2.2 S3, revision 1

The security level is an enumeration. S3 creates a variable with the appropriate textual representation for its value.

Necessary S3:

```
1. [securityLevel] = ( `[INDEX]{3}` == 1 ? 'noAuthNoPriv' : ( `[INDEX]{3}` == 2
2. ? 'authNoPriv' : ( `[INDEX]{3}` == 3 ? 'authPriv' : '(unknown)' ) ) );
```

Name:

```
1. 'Group '( $s `[INDEX]{0}` )( ($s `[INDEX]{1}`) ?? (' matching '($s `[INDEX]{1}`)) : '')
2. 'using'( `[INDEX]{2}` ? `v[INDEX]{2}` : 'any version' ) ` with [securityLevel], providing `
3. ( $s `[INDEX]{4}` )
```

Evaluates to:

Group grpcomm1 uses any version with noAuthNoPriv, providing read

S3, revision 2

The first "Name" S3 is too long and is not very readable. In the second revision, S3 assigns various parts of the name to variables and links the names at the end.

Necessary S3:

```
1. [groupName] = ( $s `[INDEX]{0}` );
2. [matchText] = ( ($s `[INDEX]{1}`) ?? (' matching '($s `[INDEX]{1}`)) : '' );
3. [versionText] = ( `[INDEX]{2}` ? `v[INDEX]{2}` : 'any version' );
4. [securityLevel] = ( `[INDEX]{3}` == 1 ? 'noAuthNoPriv' : ( `[INDEX]{3}` == 2 ? 'authNoPriv' :
5. ( `[INDEX]{3}` == 3 ? 'authPriv' : '(unknown)' ) ) );
6. [processingText] = ( $s `[INDEX]{4}` );
```

Name:

```
`Group [groupName][matchText] using [versionText] with [securityLevel],
providing [processingText]`
```

Evaluates to:

```
Group grpcomm1 using any version with noAuthNoPriv, providing read
```

The end result is the same, but the name is easier to read and understand. The difference is that each component S3 is broken up into separate variables which allows the name to be a single interpolated string.

27.5 Context

S3 evaluates OIDs. Evaluation must take place in the context of a certain SNMP agent.

S3 is used at the SNMP discover time for a particular device. All S3 statements are executed in the context of that device, meaning that every time an OID is encountered, the device is queried for its value, and that value is returned to the S3 statement.

27.5.1 Object Context

The main purpose of S3 is to define ways to describe specific objects from a generic set of rules. As the SNMP discover script goes through the possible object types, for each one, it generates a list of individual object indexes for that object type. The rules for that object type are applied to each index that it encounters to generate a unique object per index.

The following SNMP Object Editor fields are evaluated, in order, for each object:

1. Variables
    - A mini S3 script generates variables for later use.
    - Variables generated:
        - (Any present)
2. Subtype expression
    - Determines the subtype of the object.
    - Variables generated:
        - [TYPE]: The numerical value of the subtype (if any).
        - [TYPE NAME]: The name of the subtype (if any).
        - [TYPE DESCRIPTION]: The description of the subtype (if any).
3. Assert expression
    - Skipped if an object does not pass the assert expression.
    - Should not define any variables.
4. Name expression
    - Uniquely identifies an object across the SNMP plugin for the device in question.
    - Should not define any variables.
5. Description expression
    - Should provide an informative description of the object. If not set then use the object type.
    - Should not define any variables.

Before any evaluation happens, the "[INDEX]" variable is set to the SNMP index of the current object. This is a vector variable; each index in the vector corresponds with each specific index key defined for this object type. Each of those could also be a vector. Because of the sequence in which the system variables are generated; S3 generates an error if a variable is used before it is generated.

27.5.2 Indicator Context

The Expression field in the SNMP Indicator Editor for a particular indicator is also an S3 expression. Indicators should not set variables. The Maximum Value Expression field is also an S3 expression.

An indicator expression and maximum value expression can use any of the variables defined above for the object, including any user-defined variables in the Variables field.

This enables an indicator definition to get around any strange indexing (including out-of-order indexing) by using S3 to assemble the OID to poll as necessary.

27.5.2.1 Synthetic Indicators

Indicator expressions are unique because they are handled in a two pass system. The SNMP plugin does not implement S3; it implements a simple mathematical library. Any OIDs present in the data that is passed to the poller must be fully expanded with their index information. To accomplish this, two passes are used on the indicator expression.

The text-conversion functions of S3 does not work in either pass.

First Pass

The first pass is an expansion pass. It expands any OID encountered by adding the full index value to the end of it. This is equivalent to having every OID end in ".[INDEX]". The result of this pass is saved and no real values should be generated.

Second Pass

The second pass then evaluates (normally) the results of the first pass. If this pass results in a numeric value, then the indicator is presumed to exist. The results of the first pass are stored for use by the poller.

It is important to use standard mathematical expressions (and not the extra S3 operations) to perform indicator expressions. These expressions must conform to normal mathematical rules (for example, you cannot have two expressions next to each other with no joining operator). These expressions may include the following operators:

```
/ + -
```

And the following grouping symbols:

```
( )
```

Note: It is possible to get around the OID expansion in the first pass. The entire results of the evaluated first pass are passed to the second pass. Any text in the first pass does not have its quotes in the second pass. Thus, an OID may be quoted and used exactly as quoted in the second pass. The whole first pass could be quoted to prevent S3 from taking any intelligent action.

Example

To have every interface report, as an indicator, the total number of interfaces on the device (multiplied by 10), the following indicator expression does not work:

```
ifNumber.0 * 10
```

"ifNumber.0" is expanded with the default index, which in this case could be ".2", for example (yielding "ifNumber.0.2", which is not the desired outcome):

```
ifNumber.0.2 * 10
```

However, the following tricks the system into accepting the OID:

```
`ifNumber.0` * 10
```

This yields:

```
ifNumber.0 * 10
```

This is the desired outcome.

27.5.2.2 Summary of the Two-Pass System

1. Pass 1
    a. OIDs are expanded.
    b. Text is unquoted.
2. Pass 2
    a. Pass 1 is evaluated.

Notes:

- Do *not* use text-conversion functions.

27.6 ASCII Table

The following lists the first 128 ASCII values and their corresponding character value.

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

28 Enable Web Status

Most web servers use Apache. The Web Status plugin enables you to monitor the Apache mod_status. The Apache mod_status module exposes a single Web page under the /server-status directory of the Web server to display basic statistics about the Apache performance. SevOne NMS supports the /server-status location.

This topic describes how to enable Web Status devices to send Web Status data to SevOne NMS. This workflow is outside of the SevOne NMS application and may not present all of the steps your network requires to enable devices to send Web Status data. If the following instructions are not applicable for your network please reference to the device manufacturer's documentation.

Related SevOne NMS workflows include the Device Manager that provides access to the New Device page and the Edit Device page where you enable the Web Status plugin for a device. You also need to upload a certificate on the Authentication Settings page if you want to use the Web Status plugin with a log on via https.

28.1 Apache mod_status and ExtendedStatus

You must turn mod_status on on each web server device. To use the mod_status statistics such as the number of accesses, the amount of traffic, and SSL/TLS cache you must also turn ExtendedStatus on. The following sections provide instructions to turn on mod_status with ExtendedStatus turned on.

28.1.1 Gentoo

To enable mod_status in Gentoo, add the STATUS option to APACHE2_OPTS in `/etc/conf.d/apache2`

```
APACHE2_OPTS="-D DEFAULT_VHOST -D INFO -D PHP5 -D STATUS"
```

To enable SSL and collect statistics on it, set -D SSL.

To configure mod_status, edit the `/etc/apache2/modules.d/00_mod_status.conf` configuration file.

Find the Location tag and configure it as follows:

```
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from (IP address or hostname of SevOne NMS)
</Location>
ExtendedStatus On
```

Enter the following command to restart Apache and apply the changes.

```
/etc/init.d/apache2 restart
```

Enter the following command to test mod_status.

```
http://your-server-here/server-status
```

28.1.2 Ubuntu

Enter the following command to enable mod_status in Ubuntu.

```
sudo /usr/sbin/a2enmod status
```

Enter the following command to enable SSL and collect statistics on it.

```
sudo /usr/sbin/a2enmod ssl
```

To configure mod_status, edit the `/etc/apache2/mods-enabled/status.conf` configuration file.

Find the Location tag and configure it as follows:

```
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from (IP address or hostname of SevOne NMS)
</Location>
ExtendedStatus On
```

Enter the following command to restart Apache and apply the changes.

```
sudo /etc/init.d/apache2 restart
```

Enter the following command to test mod_status.

```
http://your-server-here/server-status
```

28.2 mod_status Statistics

The Web Status plugin collects all Apache statistics even if they do not appear to be supported.

28.2.1 ExtendedStatus On - (Recommended)

When you turn ExtendedStatus on, the following statistics are available.

- Server Uptime - The uptime of the entire server. Note: This is different from the ServerUptime that displays when ExtendedStatus is off which is just the uptime of the Apache process.
- Total Accesses - The number of requests sent to Apache.
- Total Traffic - The amount of traffic that Apache has sent/received.
- CPU Usage - This is a bit misleading because it is the result of the POSIX function times, which returns statistics for a single process only. Apache usually runs as a multi-process application. From the perspective of a single process, this is a counter value.
- Requests Per Second - The average number of requests per second (averaged over the life of the process).
- Bytes Per Second - The average number of bytes per second (averaged over the life of the process).
- Bytes Per Request - The average number of bytes per request (averaged over the life of the process).
- Number of Busy Workers - The number of threads that are currently serving pages in some capacity.
- Number of Idle Workers - The number of threads that are doing nothing but wait for a request to come in.

ExtendedStatus Off

(Not Recommended) If you choose to leave ExtendedStatus off, this puts the least load on Apache and results in the display of the following basic information about the Apache server itself plus a visual representation of the threads and a warning about the need to turn ExtendedStatus on.

- Server Uptime - How long the Apache server has been running.
- Number of Busy Workers - The number of threads that are currently serving pages in some capacity.
- Number of Idle Workers - The number of threads that are doing nothing but wait for a request to come in.

28.3 SSL

Apache uses a multi-process model, in which all requests are not handled by the same process. This causes the SSL session information to be lost when a client makes multiple requests. Multiple SSL handshakes cause considerable overhead on the webserver and the client. To avoid this, SSL session information must be stored in an inter-process session cache to allow all the

processes to have access to handshake information. There are two cache types: SHMCB stores the cache in shared memory as a cyclic buffer and DBM stores the cache as a DBM hashfile on the local disk.

28.3.1 SHMCB

If the Cache type is set to SHMCB, then the following statistics are available for the Web Server plugin to collect.

- Current Sessions
- Shared Memory - The amount of memory allocated to the cache.
- Subcaches - The number of subcaches.
- Indexes Per Subcache - The number of indexes per subcache.
- Index Usage - The percentage of the index used.
- Cache Usage - The percentage of the cache used.
- Total Sessions Stored Since Starting - The count of the number of sessions ever stored.
- Total Sessions Expired Since Starting - The count of the number of sessions ever expired.
- Total (Pre-expiry) Sessions Scrolled Out of the Cache - The number of sessions that have been scrolled out of the cache before they expired.
- Total Retrieves Since Starting: Hits - The number of times a session was successfully retrieved.
- Total Retrieves Since Starting: Misses - The number of times that a session was not retrieved because the session was not there.
- Total Removes Since Starting: Hits - The number of times that a session was removed successfully.
- Total Removes Since Starting: Misses - The number of times a session was not removed because the session was not there.

28.3.2 DBM

The DBM cache type provides the following statistics.

- Current Sessions
- Maximum Cache Size - Displays the maximum allowed cache size (can be unlimited).
- Current Cache Size - Displays the current size of the cache.
- Average Session Size - Displays the average size of a session.

28.4 Web Status Plugin Statistics

The Web Status plugin collects the following statistics.

- Accesses - The number of requests sent to Apache.
- Availability of the Page - Displays whether or not SevOne NMS could access the page.
- Average Request Size - The average number of bytes per request. Note: This is an average over the life of the process which is an unknown period of time, so you should avoid using this statistic.
- Average Transfer Speed -The average number of bytes per second. Note: This is an average over the life of the process which is an unknown period of time, so you should avoid using this statistic.
- Cache Usage - The percentage of the cache used.
- Current Sessions
- Extended Status On - Displays 1 (one) if ExtendedStatus is on or displays 0 (zero) if ExtendedStatus is off.
- Index Usage - The percentage of the index used.
- Indexes Per Subcache - The number of indexes per subcache.
- Removes Hit - The number of times a session was removed successfully.
- Removes Missed - The number of times a session was not retrieved because the session was not there.
- Retrieves Hit - The number of times a session was successfully retrieved.
- Retrieves Missed - The number of times that a session was not retrieved because the session was not there.
- Sessions Expired - The count of the number of sessions ever expired.
- Sessions Scrolled - The number of sessions that have been scrolled out of the cache before they expired.
- Sessions Stored - The count of the number of sessions ever stored.
- Shared Memory - The amount of memory allocated to the cache.
- Subcaches - The number of subcaches.
- Traffic - The amount of traffic that Apache has sent/received. Note: This is an average over the life of the process which is an unknown period of time, so you should avoid using this statistic.
- Workers Busy - The number of threads that are currently serving pages in some capacity.
- Workers Busy Percent - The percentage of busy workers.
- Workers Idle - The number of threads that are doing nothing but wait for a request to come in.
- Workers Idle Percent - The percentage of idle workers.

- **Workers Total** - The total number of workers.

28.4.1 Recommendations

You can define the following policies from the Policy Browser to manage the Web status monitoring of your network.

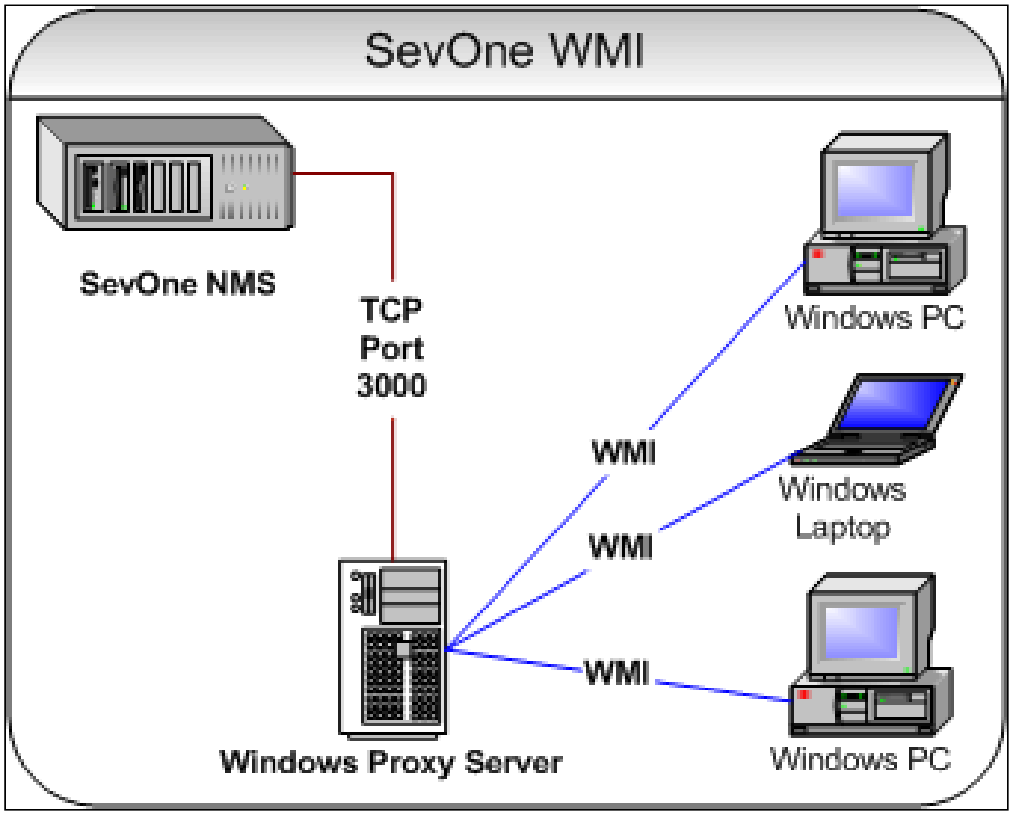
- Create a policy to alert when ExtendedStatus On is 0. If Extended Status On is equal to zero, the Web Status plugin is not being sent the most complete set of Apache data from the device. See the sections in this chapter above for details.
- Create a policy to alert when Workers Busy Percent is greater than 80%. If more than 80% of a device workers are busy, you should increase the number of available worker or this could indicate there is a problem with an application hosted on the web server.
- Create a policy to alert when Cache Usage is over 80%. The cache should be expanded.
- Create a policy to alert when Index Usage is over 80%. The cache should be expanded.

29 Enable WMI

The WMI plugin polls WMI data such as: CPU usage, ASP.NET, hard drive usage, and memory usage monitors. WMI data appears throughout the application in Instant Graphs, TopN Reports, thresholds, and other workflows. You can mix match WMI and SNMP to create logical thresholds. This topic describes workflows outside of the SevOne NMS application and may not present all of the steps your network requires to enable devices to send WMI data. If the following instructions are not applicable for your network please reference the device manufacturer's documentation.

There are several steps to set up your network to enable SevOne NMS to monitor WMI data from devices for which you enable the WMI plugin.

1. Set up the WMI proxy server to communicate with SevOne NMS (this chapter).
2. Set up your network's Windows devices to communicate with the WMI proxy server (this chapter).
3. Set up the WMI plugin for each WMI device from the New Device page and the Edit Device page.
4. Enable the WMI object types for the WMI plugin to poll from the Object Types page.



29.1 Set Up WMI Proxy Servers

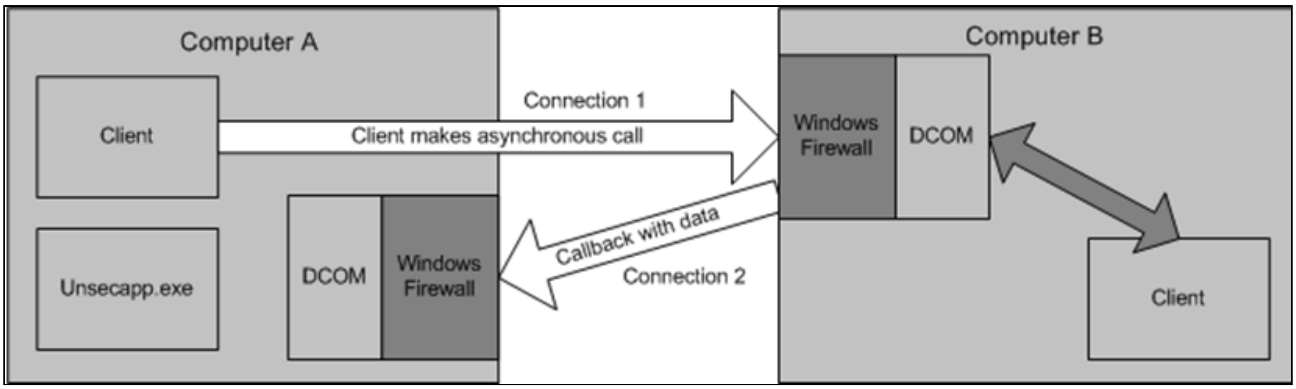
The first step to set up SevOne NMS to monitor WMI data is to create a WMI proxy server. SevOne NMS provides a Windows service for you to install on the Windows machine that you want to act as the proxy server to perform WMI queries. The WMI Proxies subtab on the Cluster Manager > **Cluster Settings** tab enables you to add, edit, and delete the WMI proxies for SevOne NMS to use.

1. In SevOne NMS, click the **Administration** menu and select **Cluster Manager** to display the Cluster Manager.
2. At the cluster level, select the **Cluster Settings** tab.
3. Select the **WMI Proxies** subtab.
4. Click the **WMI Proxy Download Installation Package** link and save the file to the proxy server.
5. If the proxy server is not running the Microsoft .NET 3.5 framework, click the **.NET 3.5 Framework Download Installation Package** link and save the file to the proxy server.
6. On the proxy server, run the **.NET 3.5 Framework setup.exe**, if needed, and then run the **SevOne NMS WMI Proxy Setup.msi**.
7. Click **Start** and select **Control Panel**.
8. Double-click **Administrative Tools**.
9. Double-click **Component Services**.

10. Double-click **Services (Local)**.
11. Right-click on **SevOne NMS WMI Proxy** and select **Start**.
12. On the proxy server, ensure that the user account is the local administrator.
13. In SevOne NMS, on the Cluster Manager > **WMI Proxies** subtab, click **Add WMI Proxy** to display the Add WMI Proxy pop-up.
14. On the Add WMI Proxy pop-up, in the **Name** field, enter the name of the proxy server.
15. In the **IP Address** field, enter the proxy server IP address.
16. In the **Port** field, enter the port for the proxy server to use to communicate with SevOne NMS (default - 3000).
17. Click **Save** on the Add WMI Proxy pop-up.
18. Repeat the previous steps to define additional WMI proxy servers.
19. Click **Save** on the WMI Proxies subtab to save the WMI Proxy cluster settings.

29.2 Set Up Windows Devices

This section describes how to enable WMI, allow remote administration, and enable the Windows device to communicate back to the WMI proxy on each Windows device from which you want to poll WMI data. WMI must establish a DCOM connection from Computer A (the SevOne NMS WMI proxy server) to Computer B (the remote computer). The following diagram shows this as Connection 1. To establish this connection, configure both the Windows Firewall and DCOM on Computer B. The configuration must be done locally on Computer B because Windows Firewall does not support remote configuration. Either execute NETSH commands to change the Group Policy settings or execute a script locally.



Computer A is the WMI Proxy Server for SevOne NMS

Computer B represents each of your network's Windows devices

29.2.1 Enable WMI

Perform the following steps on each Windows device to start the WMI service.

1. Click **Start** and select **Control Panel**.
2. Double-click **Administration Tools**.
3. Double-click **Services**.
4. Scroll down to WMI Performance Adapter, right-click **WMI Performance Adapter** and select **Start**.
5. Right-click **WMI Performance Adapter** and select **Properties**.
6. Select the **General** tab.
7. Click the **Startup Type** drop-down and select **Automatic** to have the service start every time the computer starts.
8. Continue to the next section to allow remote administration on each Windows device (create Connection 1 in the diagram).

29.2.2 Allow Remote Administration

On each Windows device, enable Allow Remote Administration to allow the Windows device to accept the communication from the WMI proxy server.

You can enter the following command.

```
netsh firewall set service RemoteAdmin enable
```

Or you can perform the following steps.

1. Click **Start** and select **Run**.
2. In the **Open** field, enter **gpedit.msc** and click **OK** to display the Group Policy page.
3. On the right side in the Local Computer Policy section, double-click **Computer Configuration**.
4. Double-click **Administrative Templates**.
5. Double-click **Network**.
6. Double-click **Network Connections**.
7. Double-click **Window Firewall**.
8. Perform one of the following steps:
    - If the computer is in the domain, double-click **Domain Profile**.
    - If the computer is not in the domain, double-click **Standard Profile**.
9. Right-click on **Windows Firewall: Allow Remote Administration Exception** and select **Properties**.
10. Select the **Enabled** option.
11. Click **OK**.
12. Click **X** to close the Group Policy page.
13. Continue to the next section to allow the Windows device to communicate with the WMI proxy server.

29.2.3 Enable Windows Devices to Communicate Back

The connection from Computer B to Computer A (Connection 2 in the diagram) is only required when the client script or application makes an asynchronous call to the remote computer. If Computer B is either a member of Workgroup or is in a different domain that is not trusted by Computer A, then Connection 2 is created as an Anonymous connection.

1. If the Windows Firewall is enabled on Computer A, run the following command on Computer A to enable Allow Remote Administration exception and to open DCOM port TCP 135 on Computer A.

    ```
    netsh firewall add portopening protocol=tcp port=135 name=DCOM_TCP135
    ```

2. Add the client application script, which contains sink for the callback to the Windows Firewall Exception List on Computer A. If the client is a script or a MMC snap-in, the sink is often Unsecapp.exe. For these connections, add `%windir%\system32\wbem\unsecapp.exe` to the Windows Firewall application exception list.

    ```
    netsh firewall add allowedprogram program=%windir%\system32\wbem\unsecapp.exe name=UNSECAPP
    ```

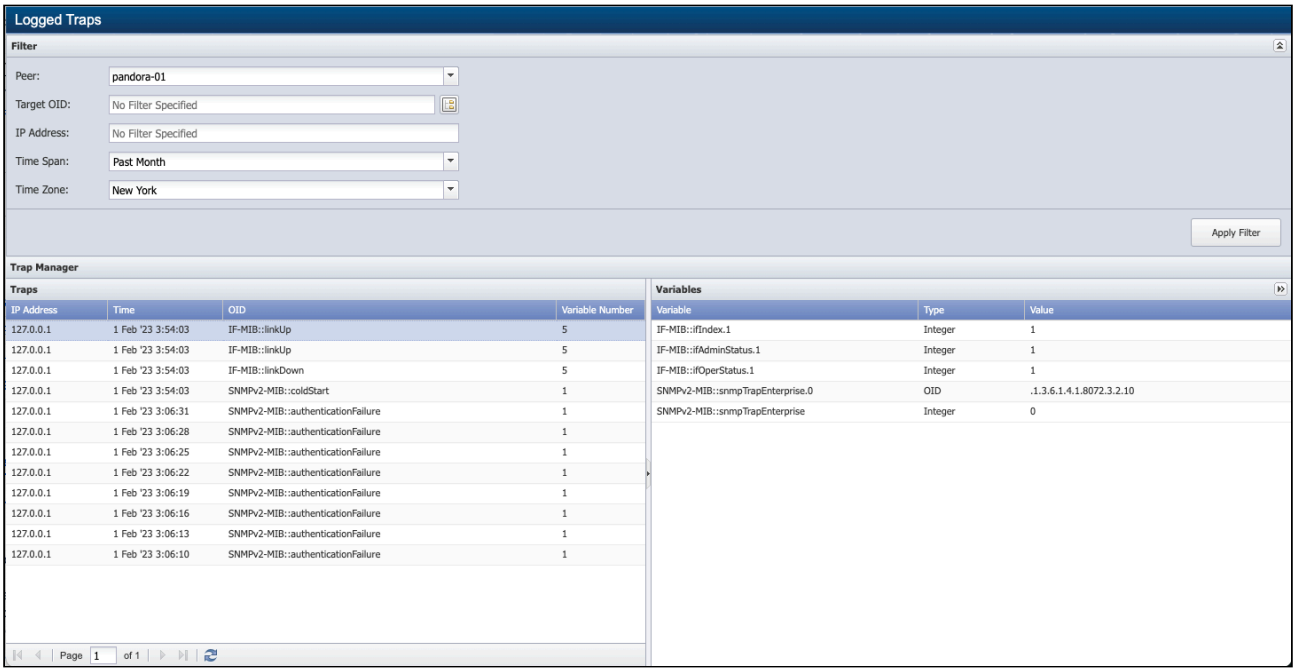
29.3 WMI Plugin and WMI Object Types

After you perform all of the steps in this topic for all Windows devices for which to monitor WMI data, turn on the WMI plugin for each device from the Edit Device page and enable the WMI object types for the WMI plugin to poll on the Object Types page.

30 Logged Traps

The Logged Traps page displays the SNMP traps SevOne NMS receives for which you define a trap event. Simple Network Management Protocol traps are an aspect of SNMP that enables a device to send information. An example of a trap trigger is when a new interface is added or a device is restarted. Trap events enable you to assign real meaning to SNMP traps and logged traps have a trap event. SevOne NMS provides starter set trap events and the Trap Event Editor enables you to define trap events that are specific to your network. The Cluster Manager > **Cluster Settings** tab enables you to define how many days to save logged traps.

To access the Logged Traps page from the navigation bar, click the **Events** menu, select **Archives**, and then select **Logged Traps**.



30.1 Filter

Filters enable you to limit the traps that appear in the list. All filters are optional and cumulative.

- Click the **Peer** drop-down and select the peer that receives the traps.
- Click the **Target OID**  to display the SNMP OID Browser or enter the name of the target OID to display traps for a specific OID.
- In the **IP Address** field, enter the IP address from which to display traps.
- Click the **Time Span** drop-down and select a time span to display traps for a specific time span.
- Click the **Time Zone** drop-down and select the time zone for the time span.
- Click **Apply Filter** button to display the traps that meet your filter criteria.

30.2 Trap Manager

30.2.1 List of Logged Traps

30.2.1.1 Traps

The logged traps list displays traps from most recent to oldest.

- **IP Address** - displays the IP address from where the trap was sent.
- **Time** - displays the time SevOne NMS received the trap.
- **OID** - displays the trap object identifier (OID) that met the conditions for the trap event.
- **Variable Number** - displays the number of variables associated with the trap.

30.2.1.2 Variables

Click on a trap to display the following information in the Variables section.

- **Variable** - displays the name of the variable.
- **Type** - displays the variable type.
- **Value** - displays the value that triggers the trap.

Min/Max recommendation for trap values

SevOne-trapd Thread Count	Processed Max	Received Max
Default = 10	1k/tps	1k/tps
Maximum = 99	1.5k/tps	4k/tps

The maximum number of traps per second (tps) that **SevOne-trapd** is able to process is 1.5k regardless of type/volume and also, regardless of configured **SevOne-trapd** thread count (The default value is 10 and maximum value is 99).
When trap count is set to its default value (=10), 1k/tps can be processed while receiving 1k/tps. If 2k/tps are sent, maximum of 1.5k/tps can be processed. Some traps are lost due to the task queue filling to its maximum, resulting in traps being discarded. This also causes **systemd-journal** to balloon in CPU usage resulting in the following error.

Error

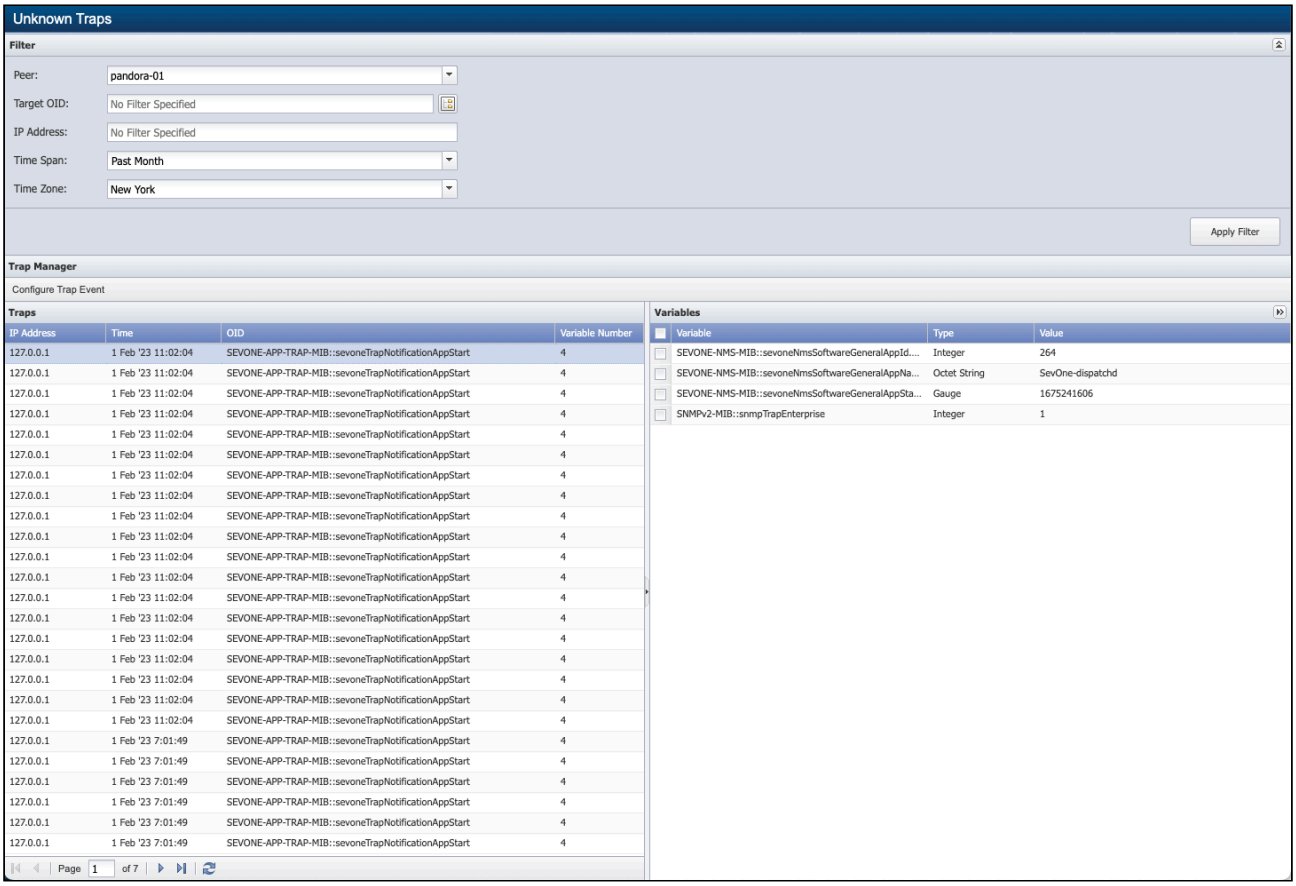
```
SevOne-trapd[2439]: MainApp::handleMessage: Failed to queue trap (task list is full).
```

When SevOne-trapd thread count is set to its maximum value (= 99), 1.5k/tps can be processed while receiving 4k/tps without overflowing the queue over time. This remains true regardless of the trap type.
There is a slight decrease in trap processing on the half-hour due to **cron** runs. This decrease is momentary and does not appear to cause a queue overload at the maximum value of 4k/tps received (1.5k/tps processed).
5k/tps received with the maximum thread count of 99 will over time cause a queue overrun. With a maximum processing rate of 1.5k/tps, the incoming traps cannot be processed fast enough resulting in incoming traps being discarded when the queue is full.

31 Unknown Traps

The Unknown Traps page displays the SNMP traps SevOne NMS receives for which you do not define a trap event. Simple Network Management Protocol traps are an aspect of SNMP that enables a device to send information. An example of a trap trigger is when a new interface is added or a device is restarted. The traps that appear on the Unknown Traps page are less meaningful than traps for which you define a trap event. Trap events enable you to assign real meaning to SNMP traps. The goal is to enable you to define trap events (either ignore, log, or alert) for the traps that are specific to your environment. The Cluster Manager > **Cluster Settings** tab enables you to define how many days to save unknown traps.

To access the Unknown Traps page from the navigation bar, click the **Events** menu, select **Archive**, and then select **Unknown Traps**.



31.1 Filter

Filters enable you to limit the traps that appear in the list. All filters are optional and cumulative.

- Click the **Peer** drop-down and select the peer that receives the traps.
- Click the **Target OID** to display the SNMP OID Browser or enter the name of the target OID to display traps for a specific OID.
- In the **IP Address** field, enter the IP address from which to display traps.
- Click the **Time Span** drop-down and select a time span to display traps for a specific time span.
- Click the **Time Zone** drop-down and select the time zone for the time span.
- Click **Apply Filter** button to display the traps that meet your filter criteria.

31.2 Trap Manager

31.2.1 Configure Trap Event

The **Configure Trap Event** button enables you to define a trap event for the unknown trap. After you define the trap event, the trap is handled in the way you specify and future trap occurrences appear on the Logged Traps page, when applicable.

1. Select a trap in the list under **Traps** (in the left pane).
2. In the right pane, select the check box for each variable to include in the trap event definition.
3. Click **Configure Trap Event** to display the Trap Event Editor with applicable fields pre-populated.

⚠ For SNMPv3 traps, if the credentials of the received traps does not match any of the entries defined in Trap v3 Receiver, you will see an error message in the OID field.
For example, your OID field will contain the following.
Decryption error (v3 securityname: MD5AES)

31.2.2 List of Unknown Traps

31.2.2.1 Traps

The unknown traps list displays traps from most recent to oldest.

- **IP Address** - displays the IP address from where the trap came.
- **Time** - displays the time SevOne NMS received the trap.
- **OID** - displays the trap's object identifier (OID) that could be used to define a trap event.
- **Variable Number** - displays the number of variables associated with the trap.

31.2.2.2 Variables

Click on a trap to display the following in the Variables section.

- **Variable** - displays the name of the variable with a check box. Select the check box to include the variable in the trap event definition.
- **Type** - displays the variable type.
- **Value** - displays the value that triggers the trap.

⚠ **Min/Max recommendation for trap values**

SevOne-trapd Thread Count	Processed Max	Received Max
Default = 10	1k/tps	1k/tps
Max = 99	1.5k/tps	4k/tps

The maximum number of traps per second (tps) that **SevOne-trapd** is able to process is 1.5k regardless of type/volume and also, regardless of configured SevOne-trapd thread count (The default value is 10 and maximum value is 99).
When trap count is set to its default value (=10), 1k/tps can be processed while receiving 1k/tps. If 2k/tps are sent, maximum of 1.5k/tps can be processed. Some traps are lost due to the task queue filling to its maximum, resulting in traps being discarded. This also causes **systemd-journal** to balloon in CPU usage resulting in the following error.

Error

```
SevOne-trapd[2439]: MainApp::handleMessage: Failed to queue trap (task list is full).
```

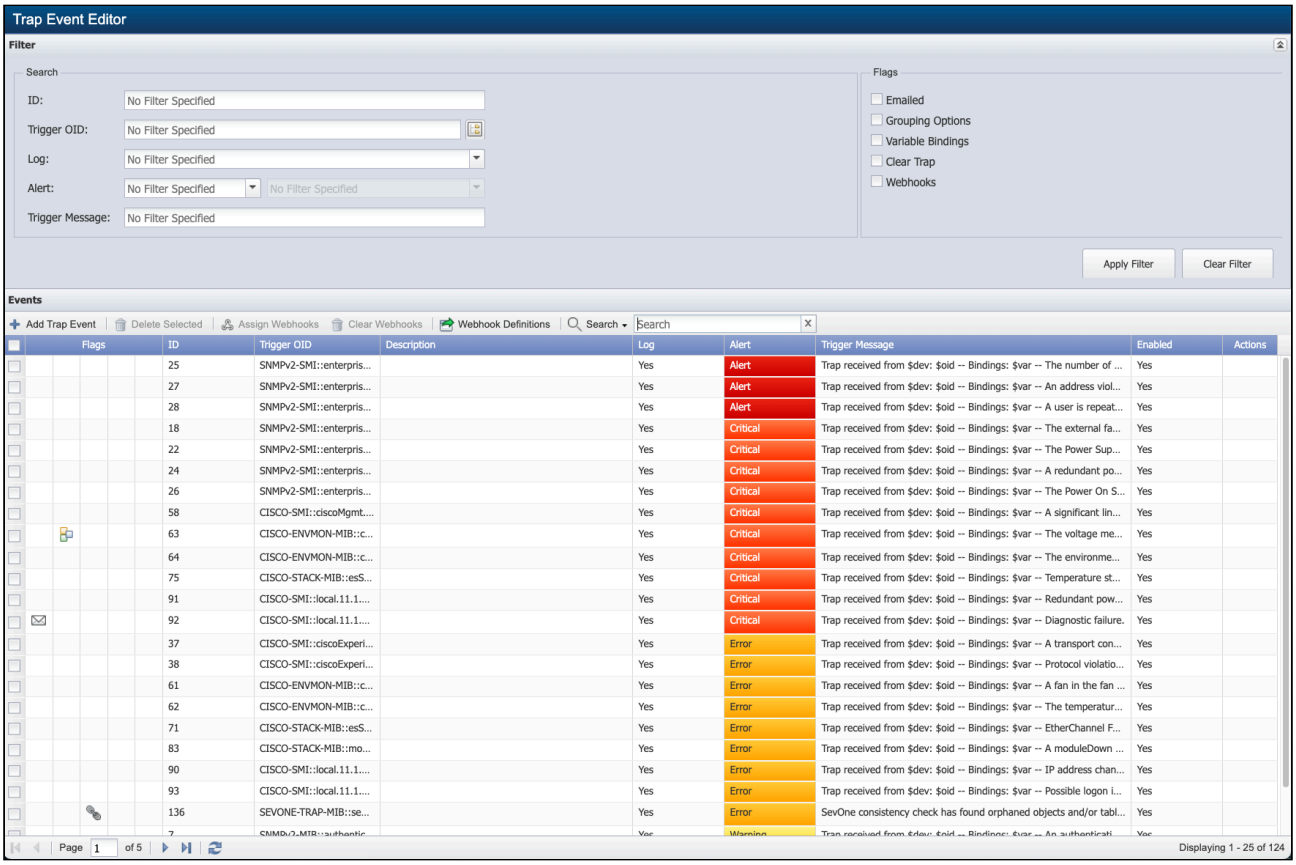
When SevOne-trapd thread count is set to its maximum value (= 99), 1.5k/tps can be processed while receiving 4k/tps without overflowing the queue over time. This remains true regardless of the trap type.
There is a slight decrease in trap processing on the half-hour due to **cron** runs. This decrease is momentary and does not appear to cause a queue overload at the maximum value of 4k/tps received (1.5k/tps processed).
5k/tps received with the maximum thread count of 99 will over time cause a queue overrun. With a maximum processing rate of 1.5k/tps, the incoming traps cannot be processed fast enough resulting in incoming traps being discarded when the queue is full.

32 Trap Event Editor

The Trap Event Editor enables you to configure how to handle traps. Traps that you associate to a trap event can appear on the Logged Traps page and traps without a trap event appear on the Unknown Traps page. SevOne NMS provides starter set trap events.

To access the Trap Event Editor from the navigation bar, click the **Events** menu, select **Configuration**, and then select **Trap Event Editor**.

Typical access to the Trap Event Editor is from the Unknown Traps page that provides a **Configure Trap Event** button. The SNMP OID Browser also provides access when you select an applicable OID and click **Configure Trap Event**.



32.1 Filter

Filters enable you to limit the trap events that appear in the list. All filters are optional and cumulative.

32.1.1 Search

1. In the **ID** field, enter an internal trap event identifier to display a specific trap event.
2. Click the **Trigger OID**  to display the SNMP OID Browser where you can select the target OID. You can enter the name of the target OID to display trap events for a specific OID.
3. Click the **Log** drop-down.
    a. Select **No Filter Specified** to display all trap events.
    b. Select **Yes** to display trap events that display applicable traps on the Logged Traps page.
    c. Select **No** to display trap events that do not display applicable traps on the *Logged Traps* page.
4. Click the **Alert** drop-down.
    a. Select **No Filter Specified** to display all trap events.
    b. Select **Yes** to display trap events that trigger an alert to appear on the Alerts page. If you select Yes, you can click the corresponding drop-down and select an alert severity to further filter the trap event list.
    c. Select **No** to display trap events that do not trigger an alert to appear on the *Alerts* page.
5. In the **Trigger Message** field, enter a portion of the trap event message to display trap events that contain the string entered in the message.

32.1.2 Flags

Select each check box to display traps that are flagged to be emailed, grouped, contain variable bindings, clear trap, and/or Webhooks. The following check boxes are available.

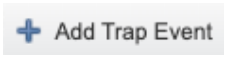
- Emailed
- Grouping Options
- Variable Bindings
- Clear Trap
- Webhooks

32.1.3 Buttons

- Click **Apply Filter** button to apply the filter settings and display the trap events that meet the filter criteria.
- Click **Clear Filter** button to remove all filters and to display all trap events in the list.
- Click on  to collapse or  to uncollapse the Filter section.

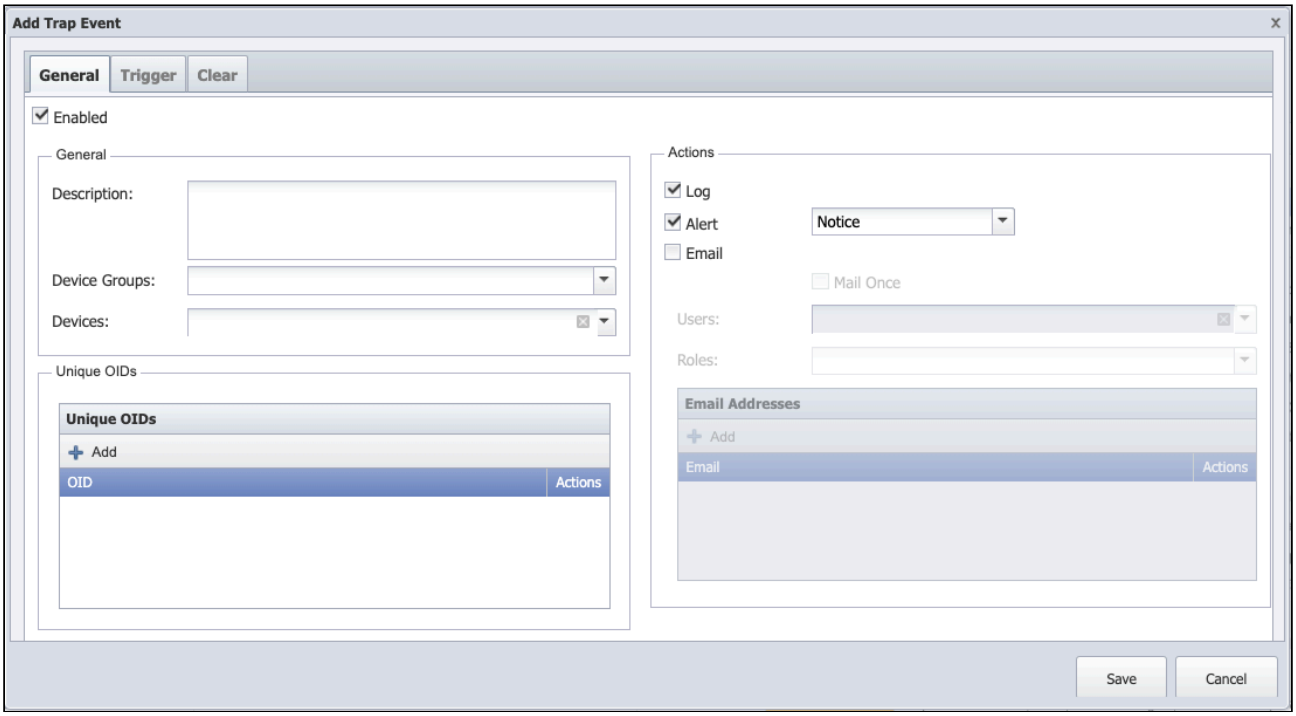
32.2 Events

32.2.1 Add / Edit Trap Event

Click  to create a new trap event. To modify an existing trap event, select a trap event and click  under **Actions** column. Or, from **Events** > **Archives** > Unknown Traps, select an unknown trap and click **Configure Trap Event** to display the Add/Edit Trap Event pop-up.

32.2.1.1 Tab 'General'

The General tab enables you to define the basic trap event settings.



1. Select the **Enabled** check box to enable the trap event. Leave clear to not apply the trap event and to display applicable traps on the Unknown Traps page.
2. Under section **General**, you can apply the trap event to device groups/device types, or devices.
    a. In field **Description**, enter description for the trap event.

b. Click the **Device Groups** drop-down and select the check box for each device group/device type to trigger the trap event.
c. Click the **Devices** drop-down and select the devices to trigger the trap event.

3. Under section **Unique OIDs**, you can designate unique OIDs to associate to the trap event.
   a. Click **Add** to add a row to the table and to add an OID.
      - Click the **OID** to display the SNMP OID Browser where you select the OID.
      - Click **Update** to save the OID with the trap event.
4. Under section **Actions**,
   a. Select the **Log** check box to display traps on the Logged Traps page. Leave clear to have traps not appear on the Logged Traps page. For an enabled trap event, when you leave this check box clear, traps that meet the trap event criteria do not appear on either the Unknown Traps page or the Logged Traps page.

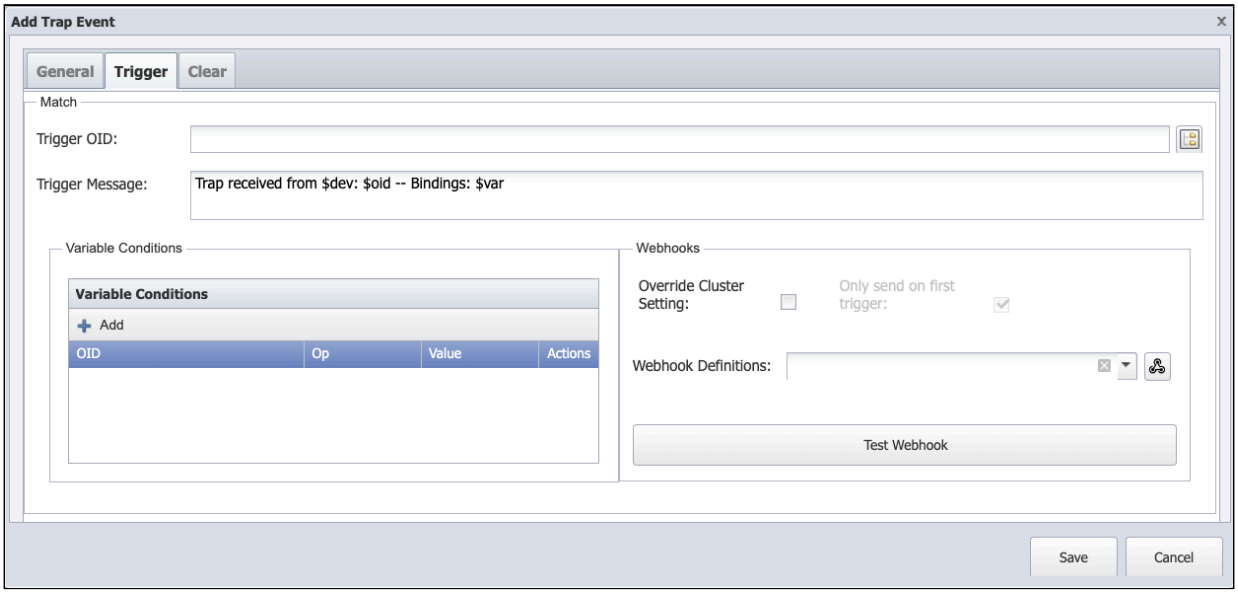
> **Examples**
>
> - For devices that send traps when traffic is denied through a firewall rule, a logged trap enables you to trace the events to a firewall to determine the cause of missed traffic.
> - Frequent but irrelevant traps such as when devices send traps each time a new IP address is leased via DHCP may not be useful.

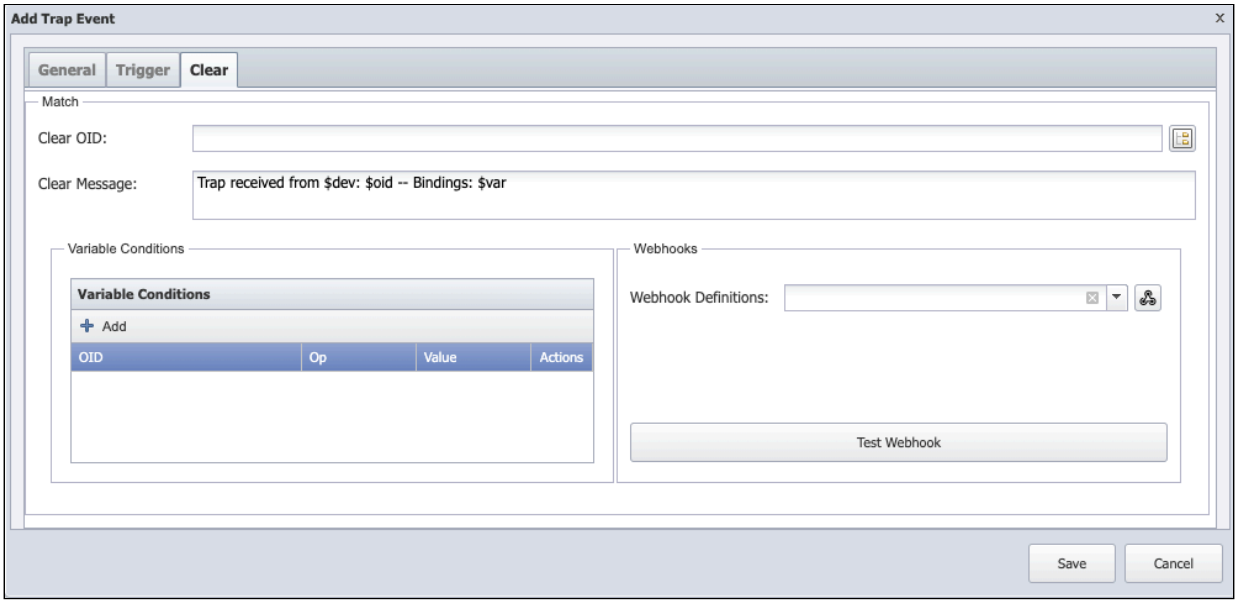
   b. Select the **Alert** check box to have the trap trigger an alert.
      - Click the drop-down and select the alert severity to display for the alert. For example, *Emergency*, *Alert*, *Critical*, *Error*, *Warning*, *Notice*, *Info*, or *Debug*.
   c. Select the **Email** check box to enable the following fields.
      - Select the **Mail Once** check box to send one email when the trap triggers the first occurrence of the trap event. All subsequent occurrences are not emailed.
      - Click the **Users** drop-down and select the users to receive an email when the trap event triggers.
      - Click the **Roles** drop-down and select the user roles to receive an email when the trap event triggers.
      - In the **Email Addresses** field, click **Add** to enter the email addresses where an email is to be sent when the trap event triggers.
   d. Click **Save As New** or **Save** to save. Click **Cancel** to cancel the add / edit of the trap event.

32.2.1.2 Tabs 'Trigger' & 'Clear'

The Trigger / Clear tabs enable you to define the conditions to trigger / clear the trap event and to define the trigger / clear messages respectively.

Trigger / Clear Trap Event





Under section **Match**, you can apply the trap event to a specific OID. A match is a logical AND option. The trap primary OID must come from the device group/device type or the device you specify to trigger the trap event. To make the trap event applicable for all device groups/device types and devices do not define Match options.

1. The **Trigger OID** / **Clear OID** provides access to the SNMP OID Browser where you select the target OID for the trap event. When you edit a trap event or you access the **Trap Event Editor** from the Unknown Traps page, this field displays the name of the OID you select. You can enter the OID name in this field if you know the OID name.
2. The **Trigger Message** / **Clear Message** field allows you to enter the message to display when this trap event is triggered. For example, *Trap received from $dev: $oid -- Bindings: $var -- The number of broadcast packets received in a second from a port is higher than the broadcast threshold.*
    a. **$dev** - to display the source device of the trap (in textual format).
    b. **$oid** - to display the trigger OID (in textual format).
    c. **$oidnum** - to display the trigger OID (in numerical format).
    d. **$var** - to display the Varbinds and respective values (in textual format).
    e. **$varnum** - to display the Varbinds and respective values (in numerical format).

    > When specifying variable OIDs (varbinds), it is helpful to review Unknown Traps. From there you can search for and identify any previous traps that have been received and the variables (varbinds) that were received with the trap.

3. Under section **Variable Conditions**, you can define the conditions for which a trap event is applicable.
    a. Click **Add** to add a row to the table and to define a new variable condition.
        - Click the **OID** to display the SNMP OID Browser where you select the trap target OID.
        - Click the **Op** drop-down and select a comparison operator.
        - In the **Value** field, enter the value that must be met to trigger the trap event.
        - Click **Update** to save the variable condition.

        > Repeat to add additional variable conditions. All variable conditions for a trap event are AND'd together.

4. Under section **Webhooks**,

    Applies to tab 'Trigger' only

Select **Override Cluster Setting** check box to override the setting in **Administration** > **Cluster Manager** > tab **Cluster Settings** > **Alerts** subtab > field **One Webhook per Alert**.

- **Only send on first trigger** check box is available only when **Override Cluster Setting** check box is selected. This allows you to override the setting configured cluster-wide. New setting is applied to the selected trap event only. When this check box is selected, it will send webhook only on the first trigger of an alert. However, when unchecked, it will send a webhook for every occurrence of an alert even if an alert already exists for that triggered trap event.

For traps, you may set field **Update Interval** from **Administration** > **Cluster Manager** > tab **Cluster Settings** > **Trap Collector** subtab. By default, Update Interval is set to 300 seconds (i.e., 5 minutes).

In *SevOne-trapd*, trap triggering list is loaded every 5 minutes based on the default value set in field **Update Interval**. When an alert is acknowledged from **Events** > **Alerts**, it does not pass through SevOne-trapd; it is now in trap triggering list's cache. If the same alert triggers again within 5 minutes after being manually acknowledged and, **Only send on first trigger** is enabled, trapd assumes that it is an incremented occurrence and ignores sending the webhook.

The lower the setting of **Administration** > **Cluster Manager** > tab **Cluster Settings** > **Trap Collector** subtab > field **Update Interval**, the lower the likelihood of webhook failing / missed. The setting of field Update Interval can affect trap webhooks.

a. Click **Webhook Definitions** drop-down and choose one or more webhook definition ids from the list. If no webhook definition ids are available or you want to create additional webhook definition ids, click  icon.
b. **Test Webhook** button provides the testing ability for the webhook definition(s) applied to the trap event. You will get a pop-up with the result for the user, including the following details. The notifications can be sent to SevOne NMS application itself.
   - **Webhook Definition ID** - returns the webhook definition id.
   - **Webhook Definition Name** - returns the webhook definition name.
   - **Ping Result** - returns the value of ping test fail or success. If success, it proceeds further.
   - **Status Code** - status code of the webhook request.
   - **Response** - when a webhook request is executed, it returns a response body.
   - **Response Error** - if webhook request fails to execute, it returns a response error.
   - **Response Header** - contains all response header values when webhook request has completed.
   - **Curl Request** - curl request has a *curl* command for every successful request.

   Click **Close** to exit.

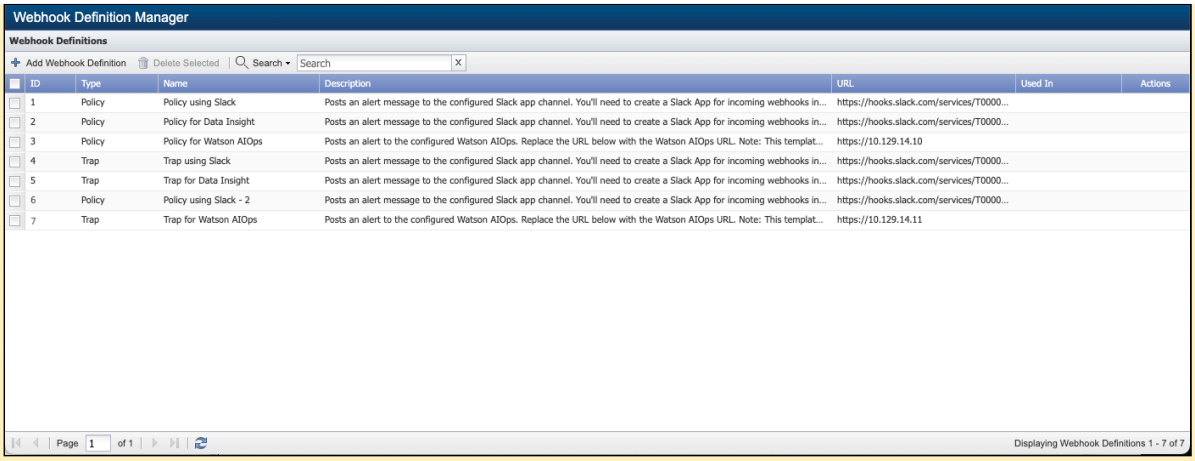
32.2.2 Delete Selected

Select one or more trap events in the list and click 

to delete the trap events selected.

32.2.3 Assign Webhooks

Before assigning webhook definitions to the trap events, you must first have webhook definitions configured. Please refer to section Webhook Definitions.
To understand *Assign Webhooks* feature, let's assume you have 7 webhook definition ids (1, 2, 3, 4, 5, 6, and 7) created. Of these 7 webhook definition ids, only 3 webhook definition ids, 4, 5, and 7 are for Type = **Trap**.

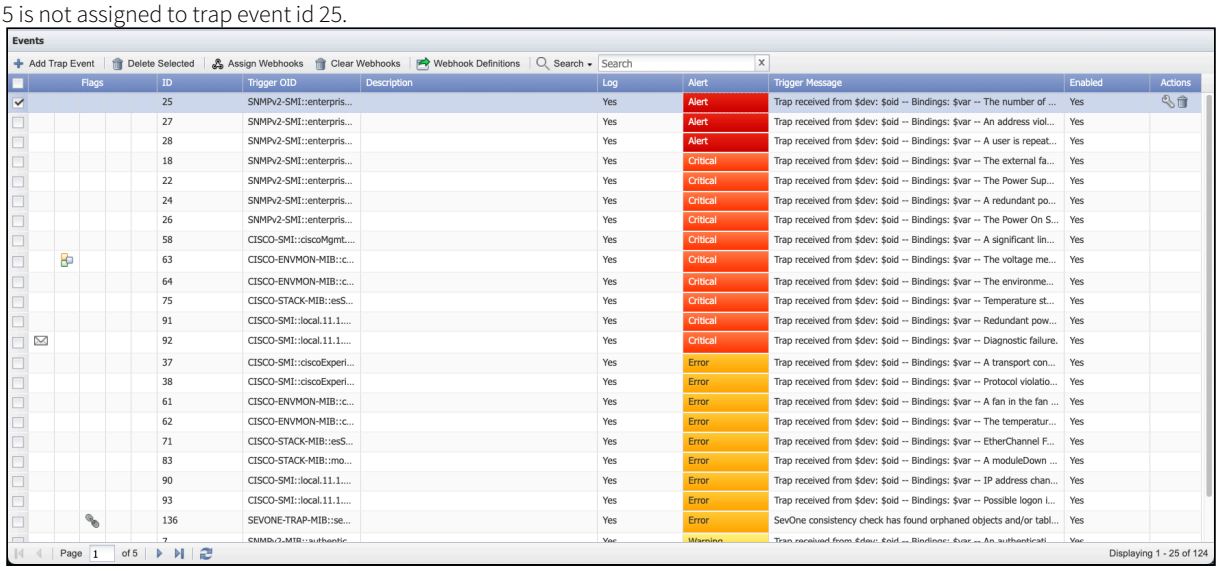


In **Used In** column, you will see *0 Trap Event(s)* for all 3 webhook definitions where Type = **Trap**. This means that these webhook definitions have not been assigned to any trap event yet.

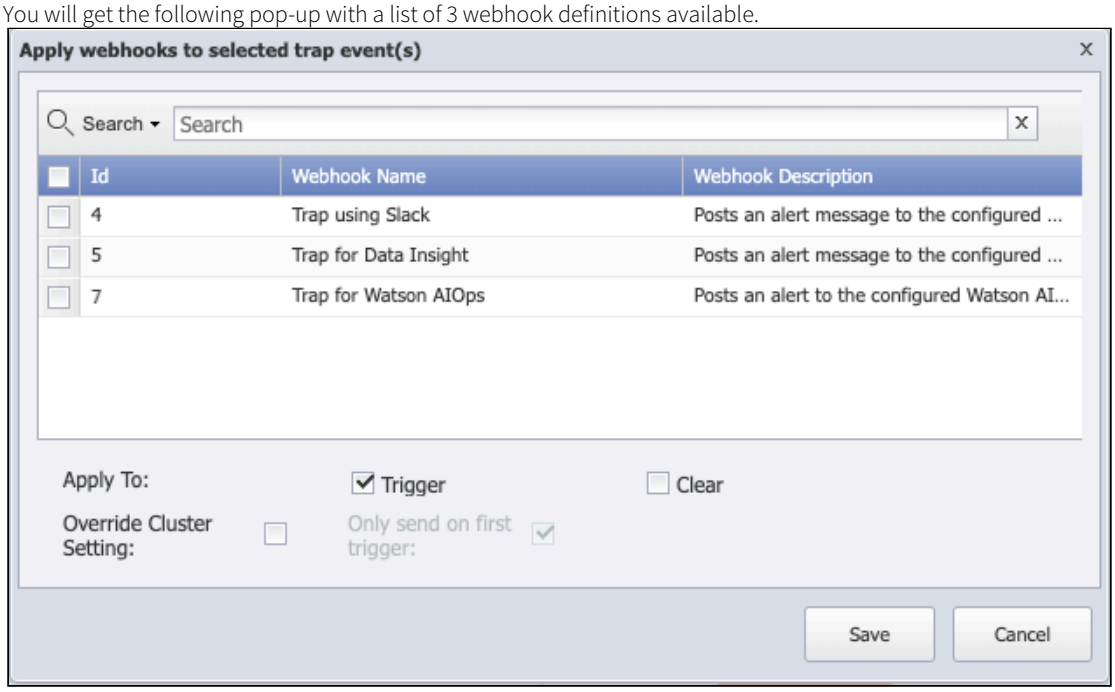
Click **Assign Webhooks** to assign webhook definitions to the trap event(s) selected. Below you will find a few scenarios.

32.2.3.1 Scenario# 1

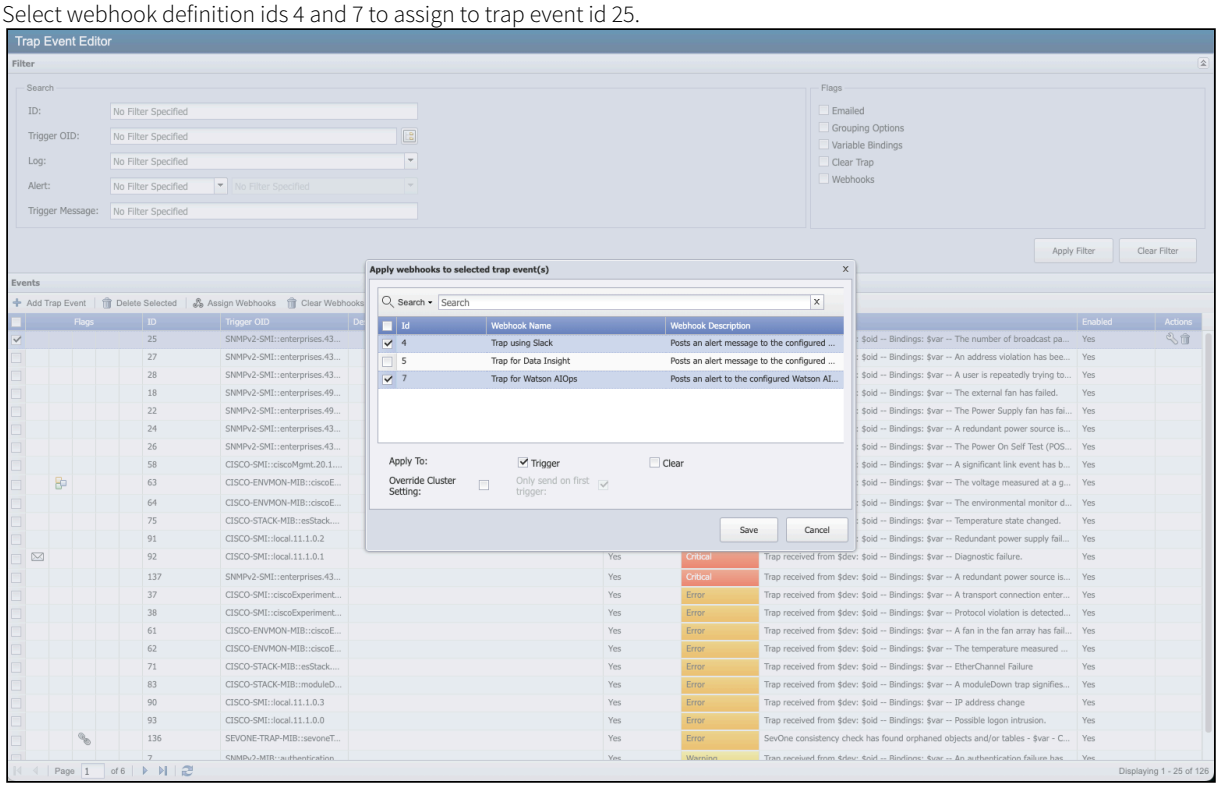
- Select trap event id 25 and click **Assign Webhooks** to assign webhook definition ids 4 and 7 to it. Webhook Definition ID 5 is not assigned to trap event id 25.



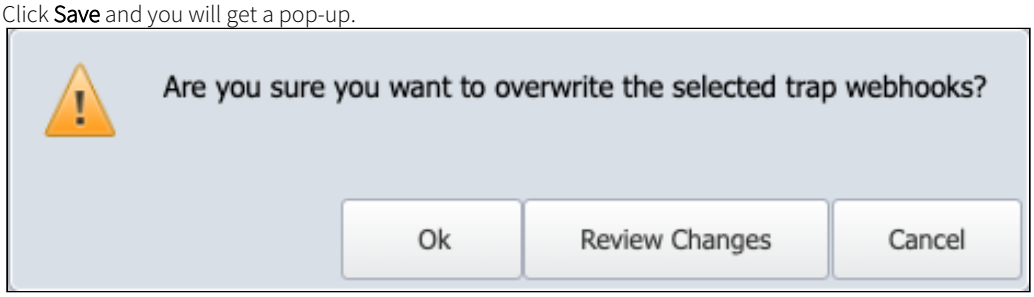
- You will get the following pop-up with a list of 3 webhook definitions available.



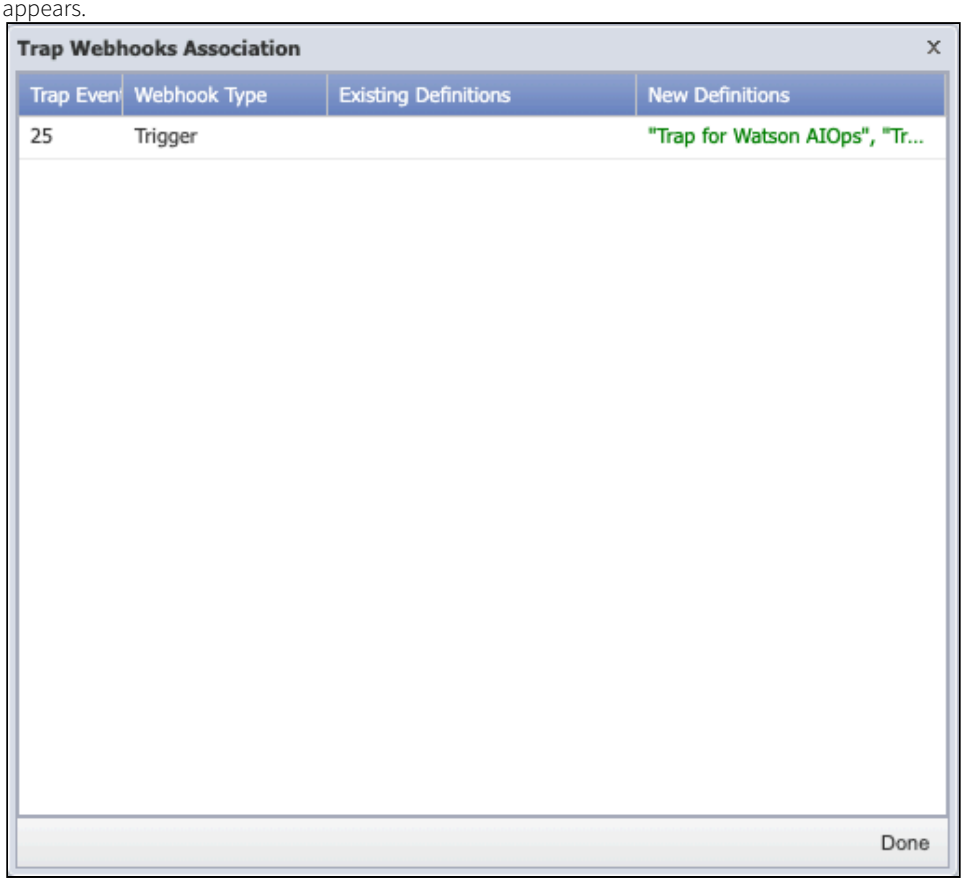
  - The **Search** field allows you to search from the list of webhook definitions available in the table below.
  - Field **Apply To** - select **Trigger** or **Clear** check box to apply the webhook definition to Trigger or Clear conditions respectively.
  - Select **Override Cluster Setting** check box to override the setting in **Administration** > **Cluster Manager** > tab **Cluster Settings** > **Alerts** subtab > field **One Webhook per Alert**.
    - **Only send on first trigger** check box is available only when **Override Cluster Setting** check box is selected. This allows you to override the setting configured cluster-wide. New setting is applied to the selected trap events only. When this check box is selected, it will send webhook only on the first trigger of an alert. However, when unchecked, it will send a webhook for every occurrence of an alert even if an alert already exists for that triggered trap event.
- Select webhook definition ids 4 and 7 to assign to trap event id 25.



- Click **Save** and you will get a pop-up.



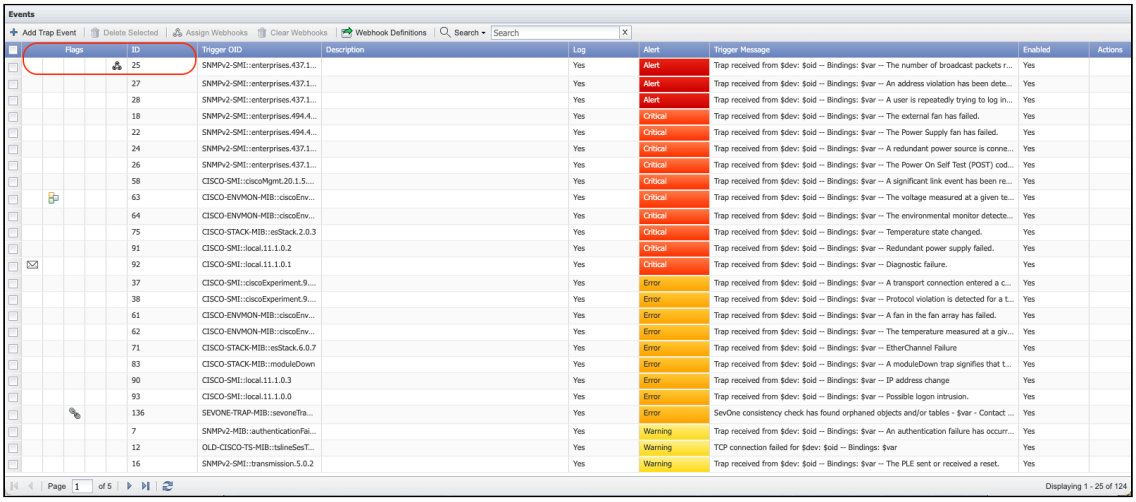
  - Click **Ok** to overwrite the webhook definitions currently assigned to trap event id 25 with webhook definition ids 4 and 7.
  - Click **Review Changes** to review the trap webhooks association before overwriting trap event id 25. A pop-up appears.



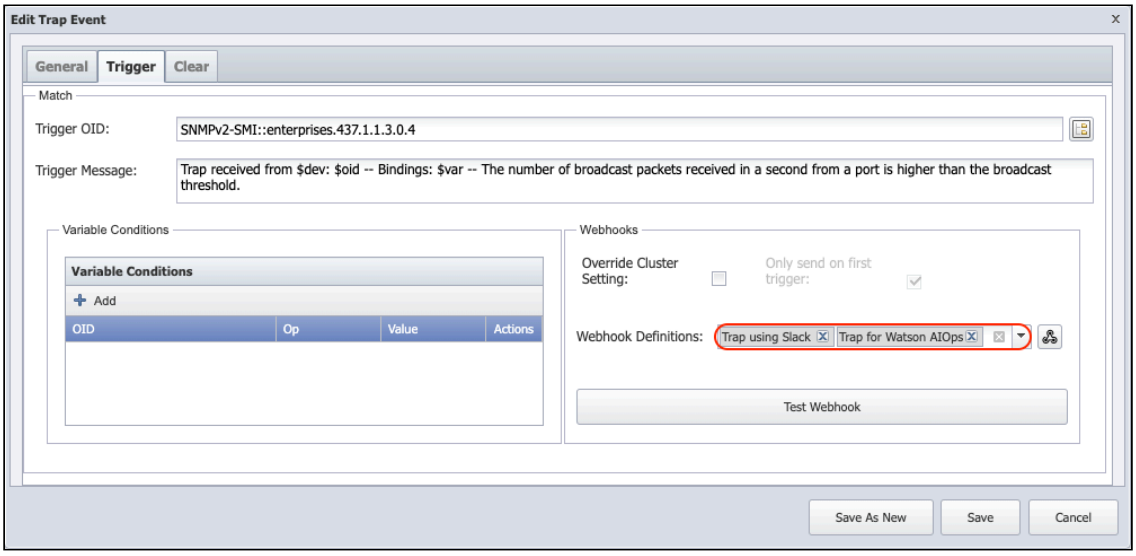
> Since this is the first time webhook definition(s) are being assigned to trap event id 25, there are no *Existing Definitions* for it.

Click **Done** after reviewing the details. If you want to continue with the assignment of the webhook definitions to the trap event(s) selected, click **Ok** to save or **Cancel** to exit.

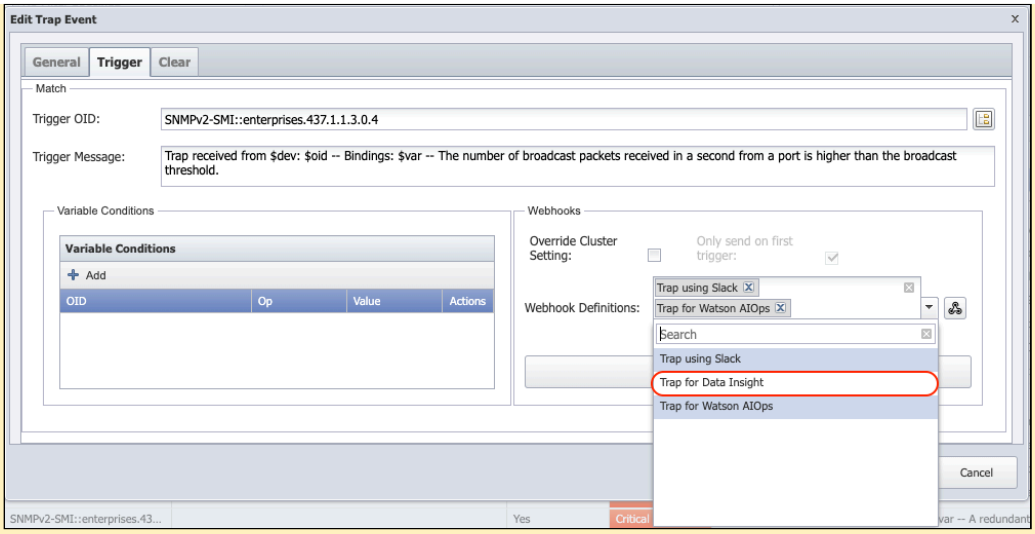
If you clicked the **Ok** button, you will see that trap event id 25 has  icon in column 5 under **Flags**. This indicates that trap event id 25 now has webhook definition ids 4 and 7 assigned to it.



To confirm this, click row with trap event id 25 > in **Actions** column click  . Choose tab **Trigger**. You will see that webhook definition ids 4 and 7 (Trap using Slack - 4 and Trap for Watson AIOps - 7 respectively) are assigned to trap event id 25.



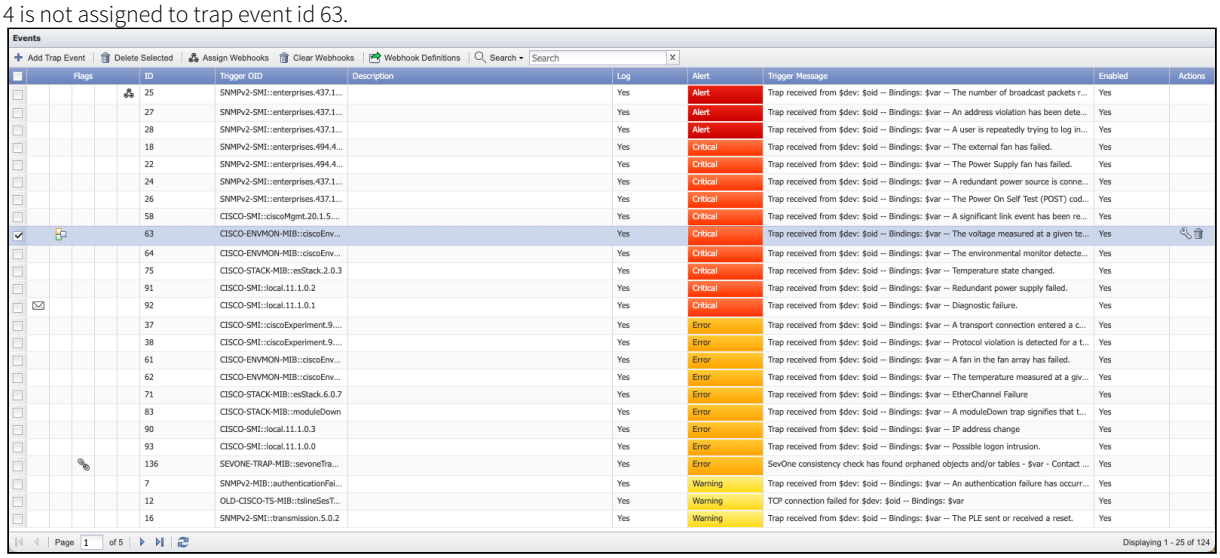
You will see that Trap Event ID 25 has Webhook Definition IDs 4 and 7 assigned to it. Webhook Definition ID 5 is available but not used.



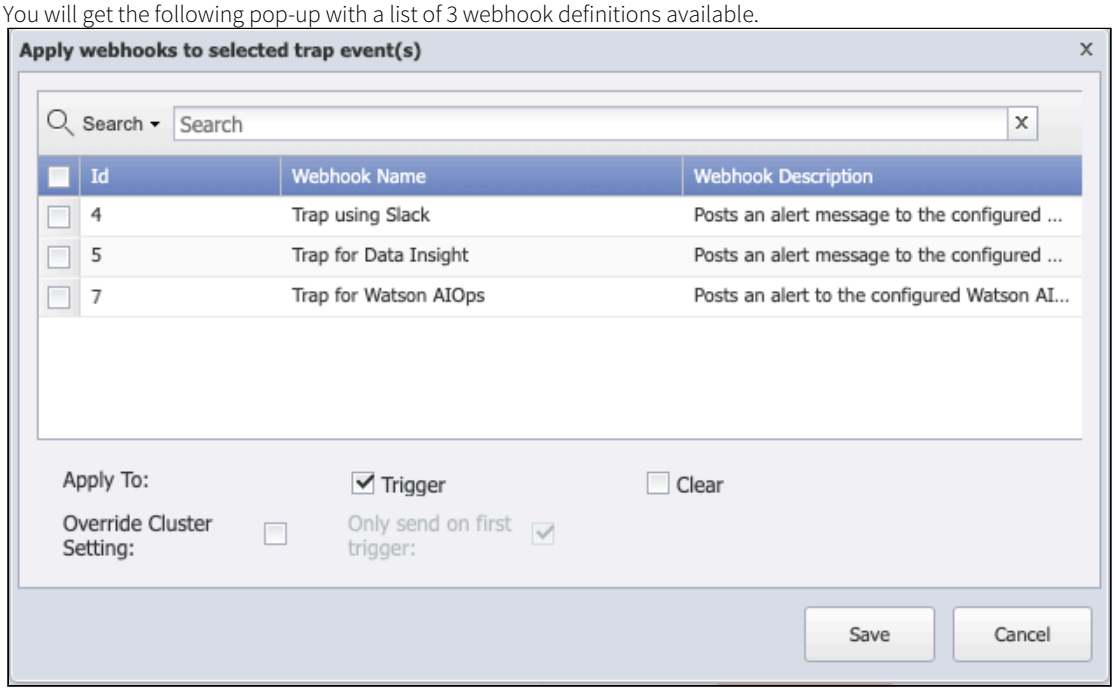
If you choose to assign Webhook Definition ID 5 to Trap Event ID 25, select **Trap for Data Insight** from the drop-down list and click the **Save** button.

32.2.3.2 Scenario# 2

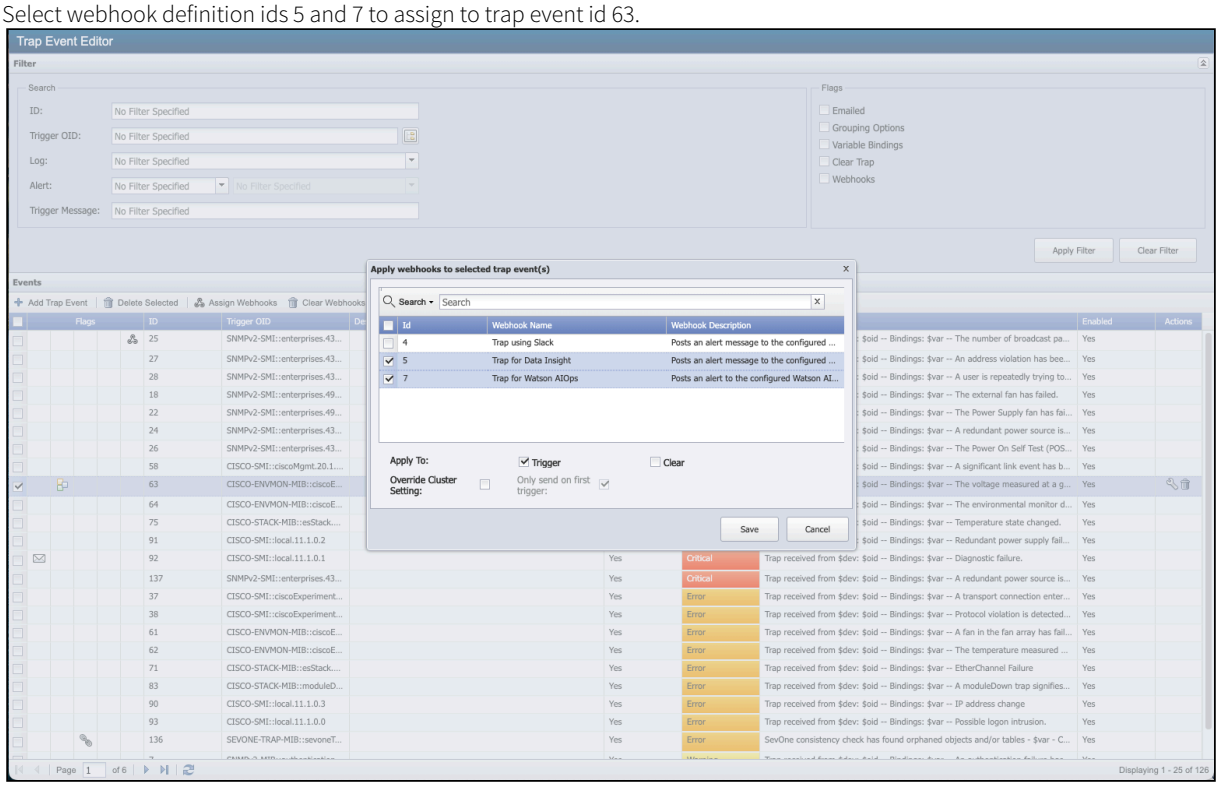
- Select trap event id 63 and click Assign Webhooks to assign webhook definition ids 5 and 7 to it. Webhook Definition ID 4 is not assigned to trap event id 63.



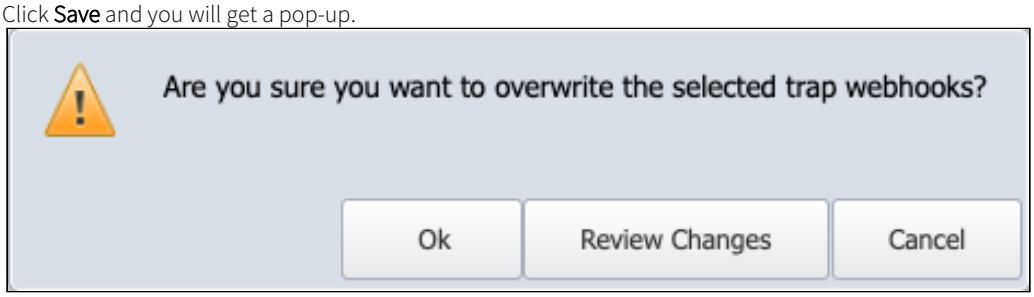
- You will get the following pop-up with a list of 3 webhook definitions available.



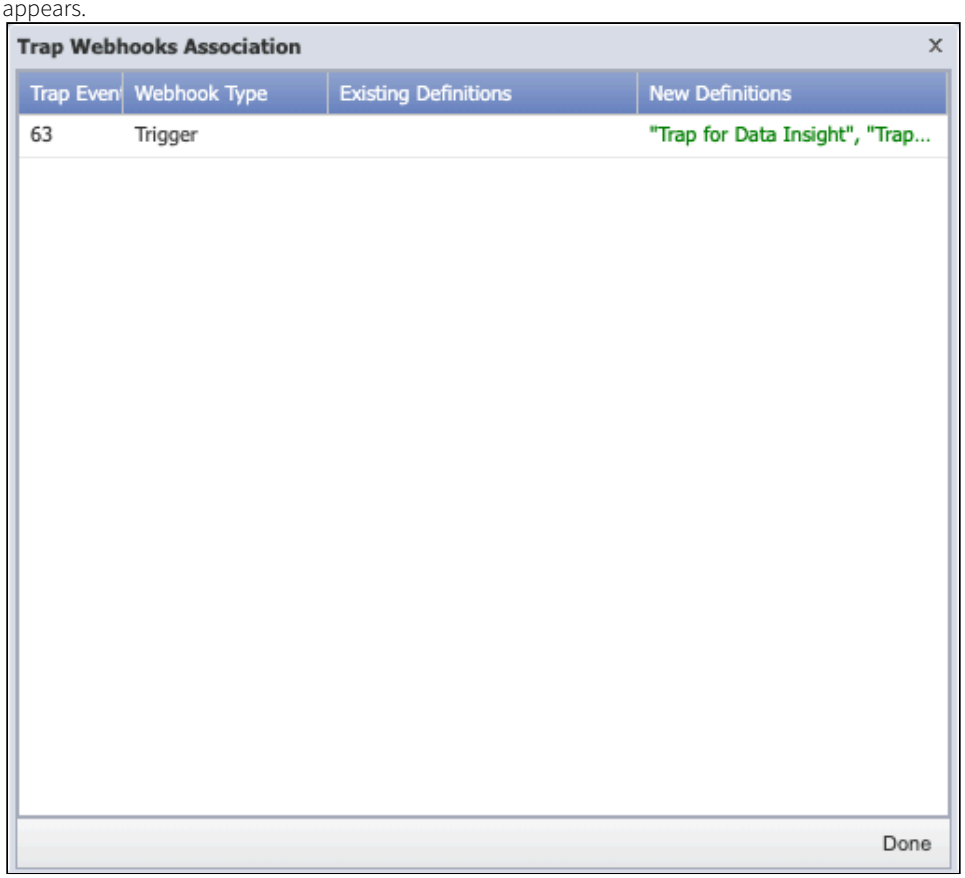
  - The **Search** field allows you to search from the list of webhook definitions available in the table below.
  - Field **Apply To** - select **Trigger** or **Clear** check box to apply the webhook definition to Trigger or Clear conditions respectively.
  - Select **Override Cluster Setting** check box to override the setting in **Administration** > **Cluster Manager** > tab **Cluster Settings** > **Alerts** subtab > field **One Webhook per Alert**.
    - **Only send on first trigger** check box is available only when **Override Cluster Setting** check box is selected. This allows you to override the setting configured cluster-wide. New setting is applied to the selected policies only. When this check box is selected, it will send webhook only on the first trigger of an alert. However, when unchecked, it will send a webhook for every occurrence of an alert even if an alert already exists for that triggered threshold.
- Select webhook definition ids 5 and 7 to assign to trap event id 63.



- Click **Save** and you will get a pop-up.



  - Click **Ok** to overwrite the webhook definitions currently assigned to trap event id 63 with webhook definition ids 5 and 7.
  - Click **Review Changes** to review the policy webhooks association before overwriting trap event id 63. A pop-up appears.

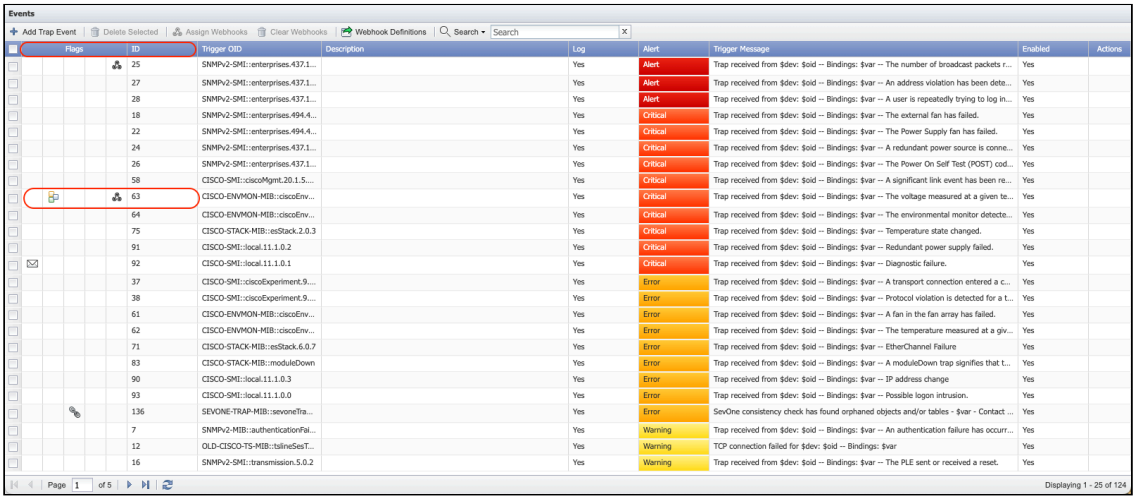


> Since this is the first time webhook definition(s) are being assigned to trap event id 63, there are no *Existing Definitions* for it.

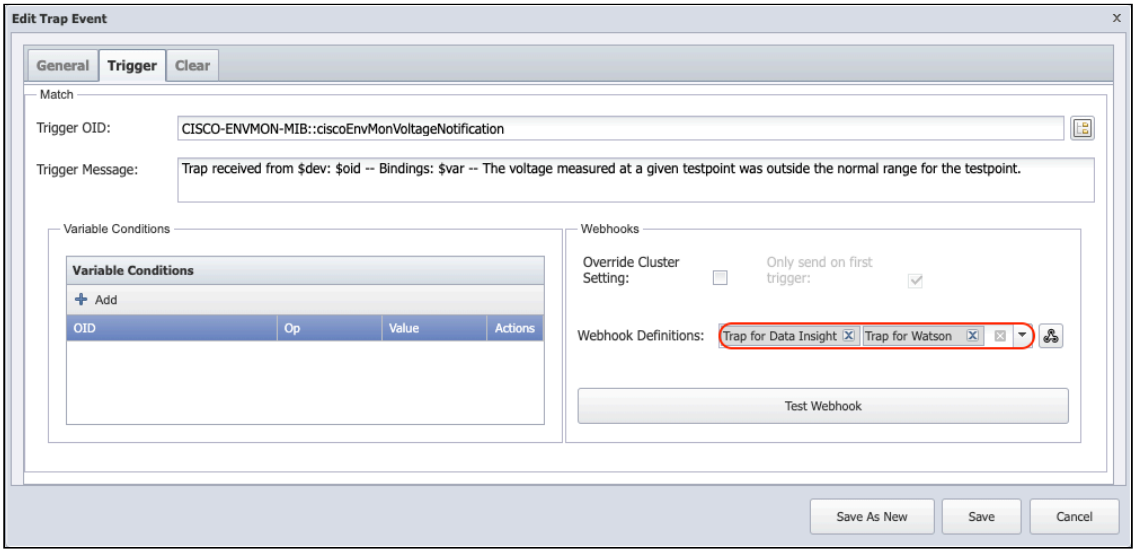
Click **Done** after reviewing the details. If you want to continue with the assignment of the webhook definitions to the trap event(s) selected, click **Ok** to save or **Cancel** to exit.

> Trap event id 63 already has  icon in column 2 under **Flags** for Grouping Options.

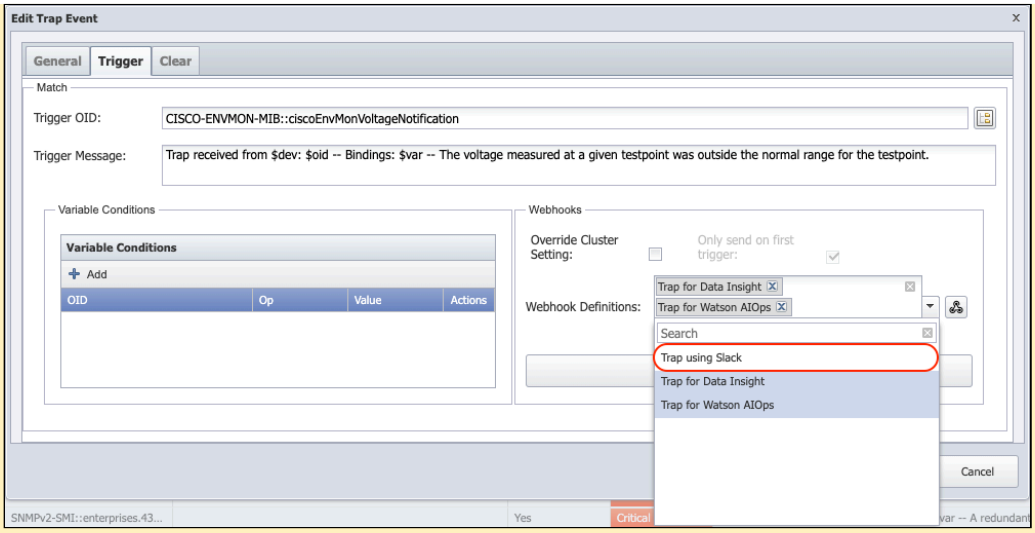
If you clicked the **Ok** button, in addition to  icon, you will also see that trap event id 63 has  icon (for webhook definition) in column 5 under **Flags**. This indicates that trap event id 63 now has webhook definition ids 5 and 7 assigned to it.



To confirm this, click row with trap event id 63 > in **Actions** column click  . Choose tab **Trigger**. You will see that webhook definition ids 5 and 7 (Trap with Data Insight and Trap for Watson AIOps respectively) are assigned to trap event id 63.



You will see that Trap Event ID 63 has Webhook Definition IDs 5 and 7 assigned to it. Webhook Definition ID 4 is available but not used.



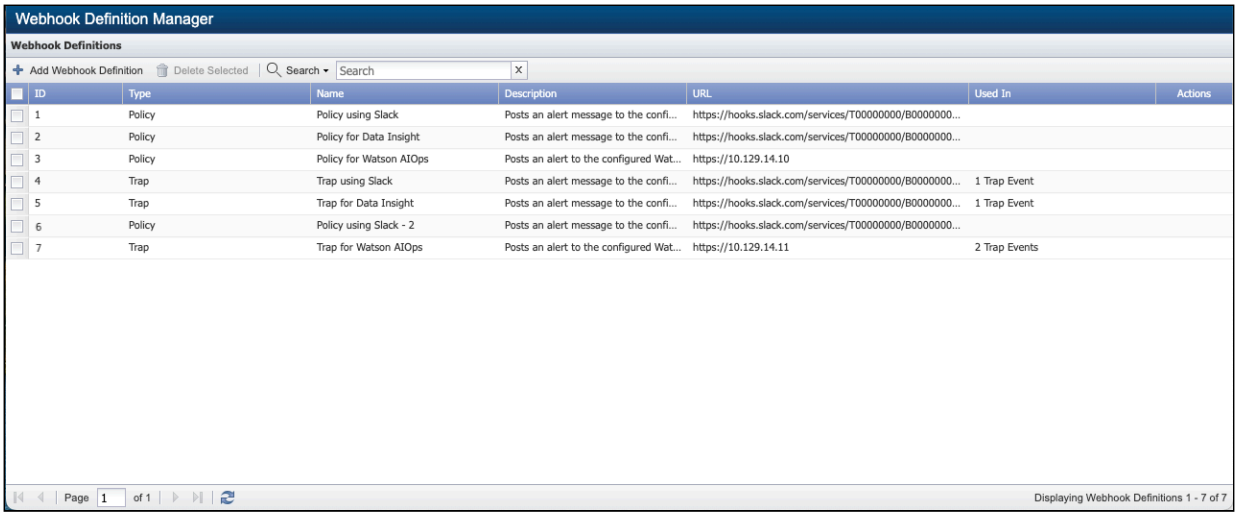
If you choose to assign Webhook Definition ID 4 to Trap Event ID 63, select **Trap using Slack** from the drop-down list and click the **Save** button.

Example

Assume that,

- Trap Event ID 25 has Webhook Definition IDs 4 and 7 assigned to it.
- Trap Event ID 63 has Webhook Definition IDs 5 and 7 assigned to it.

Based on this, **Events** > **Configure** > **Webhook Definition Manager** will appear as the following.



where,

- Webhook Definition ID 4 has *1 Trap Event* in column **Used In**. This is because only one trap event, 25, has been assigned to this ID.
- Webhook Definition ID 5 has *1 Trap Event* in column **Used In**. This is because only one trap event, 63, has been assigned to this ID.

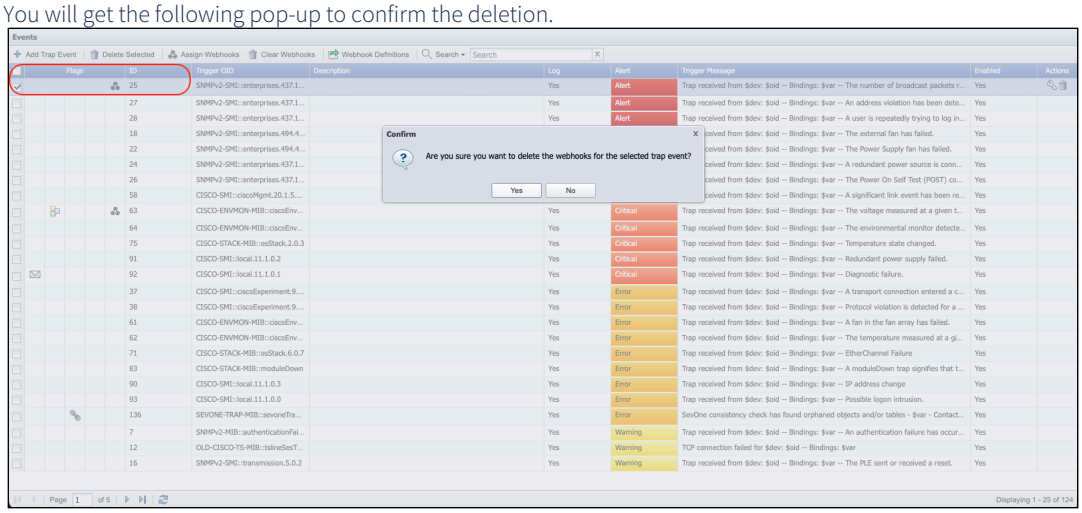
- Webhook Definition ID 7 has *2 Trap Events* in column **Used In**. This is because 2 trap events, 25 and 63, have been assigned to this ID.

32.2.4 Clear Webhooks

1. Select one or more trap event ids where webhook(s) are assigned.
2. Click Clear Webhooks to remove the webhook(s) assigned to the selected trap event id(s).

Example

- Select trap event id 25 as webhook definition ids 4 and 7 are assigned to it.
- Click Clear Webhooks .
- You will get the following pop-up to confirm the deletion.



- Click **Yes** to clear the webhooks. Otherwise, click **No**.

> If both trap event ids 25 and 63 were selected and you clicked Clear Webhooks , webhook definition ids assigned to both trap event ids 25 and 63 will get deleted.

32.2.5 Webhook Definitions

Click Webhook Definitions to create / configure, modify, or delete webhook definitions. For details, please refer to Webhook Destination Manager.

32.2.6 Search

The search capability allows user to search the table for the word enter in the field.

32.2.7 List of Trap Events

Each trap has a primary OID that designates the trap type. Each trap event has a target OID. When the trap primary OID matches the trap event target OID and any trap event variable conditions you define, the trap triggers the trap event. The list of trap events displays the following information.

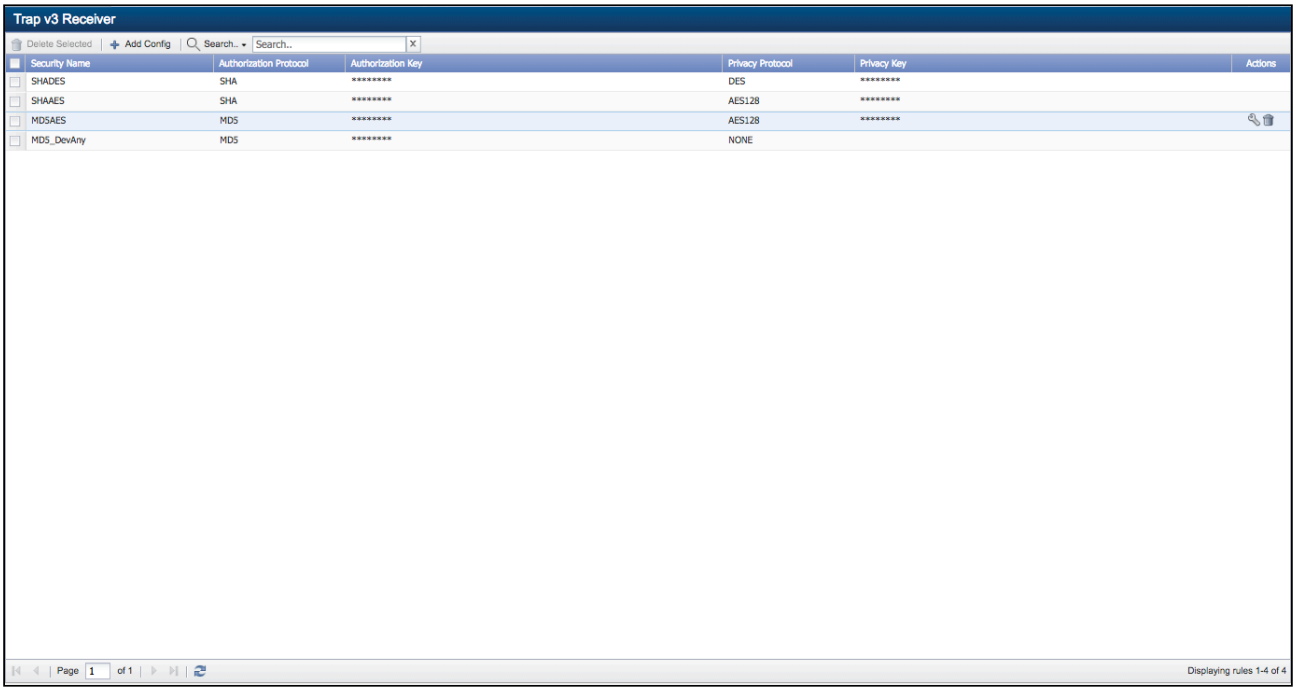
1. **Flags** - can display the following flags. There are 5 columns under Flags.

   a. column 1 - displays  when trap event triggers an email.
   b. column 2 - displays  when trap event applies to specific Device Groups / Device Types.
   c. column 3 - displays  when trap has variable conditions.
   d. column 4 - displays  when trap has clear condition(s) assigned to it.
   e. column 5 - displays  when trap has webhook definition id(s) assigned to it.
2. **ID** – displays the internal identifier for the trap event which is helpful for API workflows.
3. **Trigger OID** - displays the resolved name of the trap event target OID.
4. **Description** - displays the general description for the selected trap event id.
5. **Log** - displays *Yes* when you define the trap event to display the trap on the Logged Traps page or displays *No* when the trap does not appear on the Logged Traps page.
6. **Alert** - displays the severity level for the alerts the trap triggers when you define the trap event to trigger an alert or displays *No* when you define the trap event to not trigger an alert.
7. **Trigger Message** - displays the message you define for the trap to display.
8. **Enabled** - displays *Yes* when the trap event is enabled or displays *No* when the trap event is disabled.

33 Trap v3 Receiver

The Trap v3 Receiver enables you to configure the user credentials for receipt of SNMP v3 traps and informs.

To access the Trap v3 Receiver from the navigation bar, click the **Events** menu, select **Configuration**, and then select **Trap v3 Receiver**.



The Trap v3 Receiver has the following tabs.

- The **Add Config** tab enables you to add new user credentials for SNMP v3 traps and informs.
- Select the check box for each user credential to mark for deletion and click **Delete Selected** tab to delete the user credential from the list.

The following controls appear in the Actions column.

- Click to edit an existing user credential.
- Click to select the user credential that you want to delete and confirm deletion.

33.1 Add Config

Perform the following steps to add a new user credentials.

1. Click **Add Config** to display a new user credential row to be created.
2. In the **Security Name** field, enter the user security name.
3. Click the **Authorization Protocol** drop-down.
    a. Select **NONE** (`usmNoAuthProtocol`) to not use an authentication method to send or receive messages.
    b. Select **MD5** (`usmHMACMD5AuthProtocol`) to use MD5 authentication protocol for messages.
    c. Select **SHA** (`usmHMACSHAAuthProtocol`) to use SHA authentication protocol for messages.
4. If you select MD5 or SHA in the previous step, in the **Authorization Key** field, enter the password for the authorization protocol that SevOne NMS requires to authenticate traps from the device.
5. Click the **Privacy Protocol** drop-down.
    a. Select **None** to not use encryption to send or receive messages.
    b. Select **DES** to use the Data Encryption Standard encryption method.
    c. Select **AES128** to use the Advanced Encryption Standard encryption method using keys sized at 128 bits.
    d. Select **AES192** to use the Advanced Encryption Standard encryption method using keys sized at 192 bits.
    e. Select **AES256** to use the Advanced Encryption Standard encryption method using keys sized at 256 bits.
6. If you select DES or AES128 or AES192 or AES256 in the previous step, in the **Privacy Key** field, enter the key for the privacy protocol that SevOne NMS requires to decrypt traps from the device.

7. Click **Update** to save.

33.2 Edit Config

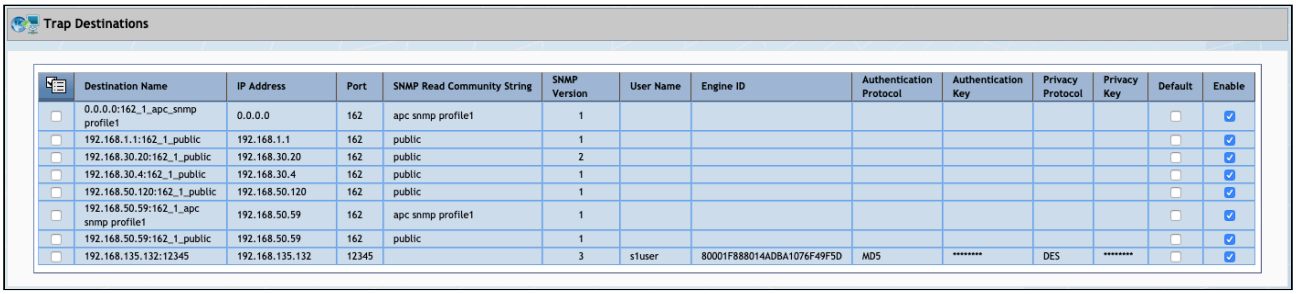
To edit an existing user credential, click  to edit a Trap v3 Receiver. Click **Update** to save.

Any additions or changes to user credentials are loaded into the Trap v3 Receiver every 5 minutes.

34 Trap Destinations

The Trap Destination page enables you to define the destinations where you want SevOne NMS to send traps. Trap destinations can be third party applications such as your company's event console or fault management system. Each trap can be sent to multiple destinations. After you define a trap destination, you associate devices to the trap destination from the Trap Destination Associations page to have devices send traps to the destinations you define here.

To access the Trap Destination page from the navigation bar, click the **Events** menu, select **Configuration**, and then select **Trap Destinations**.



34.1 Manage Trap Destinations

The Trap Destinations pop-up enables you to manage trap destinations.

1. Click  and select **Add New Destination** or select the check box for a trap destination, click  , and then select **Edit Selected** to display the Trap Destination Settings pop-up.
2. In the **Destination Name** field, enter the trap destination name.
3. In the **IP Address** field, enter the IP address of the trap destination device.
4. In the **Port Number** field, enter the port number to which to send the trap.
5. Click the **SNMP Version** drop-down and select an SNMP version. For example, choose 1 for SNMPv1, 2 for SNMPv2, and 3 for SNMPv3.
    a. If **SNMPv1** or **SNMPv2** are chosen from the drop-down, in the **SNMP Read Community String** field, enter the read community string SevOne NMS needs to authenticate onto the device.
    b. If **SNMPv3** is chosen from the drop-down, the following fields are available.
        i. In the **User Name** field, enter a username.
        ii. In **Engine ID** field, enter the engine id which uniquely identifies the host. Enter the URL for the SevOne appliance (i.e. the trap destination device) into your web browser. On the navigation bar, click the **Administration** menu, select **Cluster Manager**, and then select **Cluster Overview** tab to obtain the Engine ID.
        iii. Click the **Authentication Protocol** drop-down and select from **NONE**, **MD5**, **SHA**, **SHA224**, **SHA256**, **SHA384**, or **SHA512**.
        iv. In the **Authentication Key** field, enter the password for the user.
        v. Click the **Privacy Protocol** drop-down and select from **NONE**, **AES**, **AES192**, **AES192C**, **AES256**, **AES256C**, **DES**, or **3DES** to encrypt the trap.

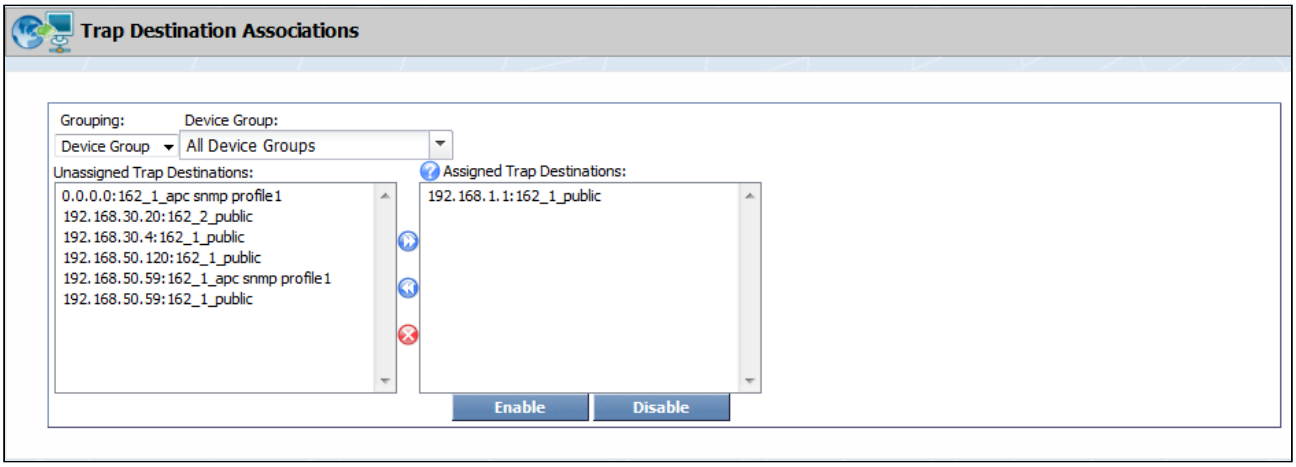
        > **IMPORTANT**
        >
        > If you are upgrading from SevOne NMS 5.7.2.x to SevOne NMS 6.3 or above, *no action is required*; trap destinations specified as **AES192** or **AES256** are automatically migrated to their Cisco equivalents, **AES192C** or **AES256C** respectively. New trap destinations against Cisco device(s) that use **AES192** or **AES256**, need to be specified as **AES192C** or **AES256C** respectively.

        vi. In the **Privacy Key** field, enter the privacy key.
6. Select the **Default** check box to send traps to the destination by default. You can designate multiple default trap destination and you can define individual thresholds and policies to not use a default destination for specific traps when you define thresholds and policies.
7. Select the **Enable** check box to enable the trap destination.
8. Click **Save**.

35 Trap Destination Associations

The Trap Destination Associations page enables you to associate the trap destinations you define on the Trap Destinations page with a device group/device type or a device. SevOne NMS sends traps from device groups/device types and devices to their associated trap destinations. Trap destinations can be applications such as your company's third party event console or fault management system.

To access the Trap Destination Associations page from the navigation bar, click the **Events** menu, select **Configuration**, and then select **Trap Destination Associations**.



35.1 Associate Devices with Trap Destinations

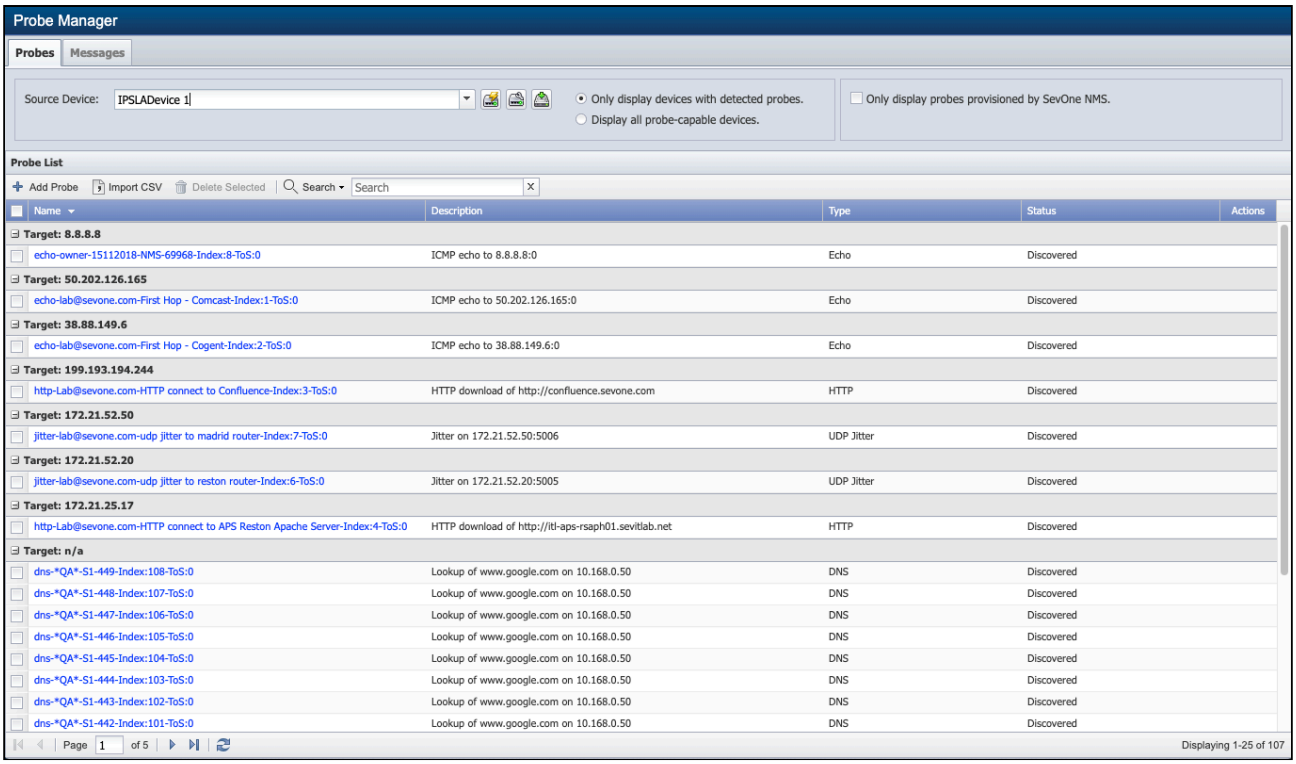
Perform the following steps to associate trap destinations with a device group/device type or a device.

1. Click the **Grouping** drop-down.
    - Select **Device Group** and then click the Device Group drop-down and select the device group/device type to which to associate trap destinations.
    - Select **Device** and then click the Device drop-down and select the device to which to associate trap destinations.
2. Move the trap destinations to associate to the device group/device type or device to the **Assigned Trap Destinations** field (use the Ctrl or Shift keys to multi-select). Traps are sent from the device group/device type or device to the trap destinations in the Assigned Trap Destinations field.
3. In the Assigned Trap Destinations field, select a trap destination and click **Disable** to save the trap association but not send traps to the destination. Disabled trap destinations appear in light text.
4. In the Assigned Trap Destinations field, select a disabled trap destination and click **Enable** to enable the association.

36 Probe Manager

The Probe Manager enables you to manage IP SLA data from the devices for which you enable the IP SLA plugin. SevOne NMS supports the following Cisco IP SLA probes; DHCP, DLSw, DNS, Echo, Ethernet Jitter, Ethernet Ping, FTP, HTTP, ICMP Jitter, RTP, TCP Connect, UDP Echo, UDP Jitter, Video, VoIP. For details, please refer to chapter IP SLA.

To access the Probe Manager from the navigation bar, click the **Applications** menu and select **Probe Manager**.



36.1 Probe List

While IP SLAs apply to two routers, each IP SLA resides on only one router. SevOne NMS can either run the IP SLA test or can provision the test to the router. The source device is the router on which the IP SLA resides.

1. Click the **Source Device** drop-down and select a device on which you enable the IP SLA plugin.
   - Click  to display a link to the Device Summary and links to report templates that are applicable for the device.
   - Click  to navigate to the Edit Device page for the source device.
   - Click  to provision the IP SLA test on the router.
2. Select one of the following options.
   - Select **Only Display Devices With Detected Probes** to display only devices on which SevOne NMS discovers at least one probe.
   - Select **Display All Probe-capable Devices** to display all devices on which you enable the IP SLA plugin.
3. Select the **Only Display Probes Provisioned by SevOne NMS** check box to display only the probes SevOne NMS provisions. Leave clear to display all probes for the source device.

The Probes list displays the following information for the probes that meet your filter criteria. Each probe has a target device whose name or IP address displays in the **Target** separator bars in the list.

- **Name** - Displays the name of the probe. Click on the name to display a link to the Object Summary and links to report templates that are applicable for the object. The probes SevOne NMS discovers have the following name convention; type, owner, and probe tag.
- **Description** - Displays the description of the probe. The probes SevOne NMS discovers have descriptions based on the probe itself.
- **Type** - Displays the type of the probe. Displays as the enumeration form from the CISCO-RTT-MON MIB.

- **Status** - Displays *Discovered* when SevOne NMS runs the test and finds the result on devices for which you enable the IP SLA plugin. Displays *Provisioned with SevOne NMS* when the device runs the test and sends the result back to SevOne NMS. Displays *To Be Deleted* if you select to delete the probe.
- Click  to view the IP SLA configuration.

36.1.1 Add Probes

Perform the following steps to add a new probe. The Import CSV button enables you to import multiple probes from a .csv file. See the Import CSV section below.

1. Click **Add Probe** to display the Add Probe pop-up.
2. On the pop-up in the **Name** field, enter the probe name. This must be unique for the device you select.
3. In the **Description** field, enter the probe description for user information only.
4. Click the **IP SLA Type** drop-down and select an IP SLA type. The type you select determines the configuration steps. IP SLA types that are not applicable for the device you select appear in light text and you cannot select those types.
    - If you select **DHCP**, perform the following configuration steps.
        i. Select one of the following options.
            - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
            - Select **Specify Target By IP Address** to enter the IP address of the target device in a text field in the next step.
        ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
        iii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
        iv. In the **Circuit ID** field, enter the Cisco Circuiter ID on its CLI. This is a hexadecimal value.
        v. In the **Remote ID** field, enter the remote ID. This is a hexadecimal value.
        vi. In the **Subnet Mask** field, enter the Cisco extension to DHCP Option 82 that allows the specification of a subnet. This is a hexadecimal value.
        vii. Click the **Source IP** drop-down and select the IP address of the source device.
    - If you select **DLSw** or **Echo**, perform the following configuration steps.
        i. Select one of the following options.
            - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
            - Select **Specify Target By IP Address** to enter the IP address of the target device in a text field in the next step.
        ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
        iii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
        iv. (Echo only) In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLAs operation (number between 0 and 255).
        v. (Echo only) Click the **Source IP** drop-down and select the IP address of the source device.
    - If you select **DNS**, perform the following configuration steps.
        i. In the **Domain** field, enter the complete valid URL of the DNS domain.
        ii. In the **Nameserver** field, enter the IP address of the name server.
        iii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
        iv. Click the **Source IP** drop-down and select the IP address of the source device.
    - If you select **Ethernet Jitter** or **Ethernet Ping**, perform the following configuration steps.
        i. In the **MPID** field, enter the destination Maintenance Point Identifier to test.
        ii. In the **Domain Name** field, enter the domain in which the destination maintenance point lies.
        iii. In the **Target VLAN** field, enter identifier of the VLAN in which the destination maintenance point lies.
        iv. In the **Target EVC** field, enter the Ethernet Virtual Connection in which the maintenance point lies.
        v. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
        vi. (Ethernet Jitter only) In the **Packet #** field, enter the number of packets to send.
        vii. (Ethernet Jitter only) In the **Interval** field, enter the number of milliseconds to be the interval between packets.
        viii. In the **CoS** field, enter the Ethernet Class of Service.
    - If you select **FTP**, perform the following configuration steps.

i. In the **URL** field, enter the complete, valid URL of the file to fetch. This must include the ftp:// part and if a user name and password are required, use the following format; username:password@webaddress.

> **Example**
>
> ftp://jerry:password@www.test.com/membersarea

ii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
iii. In the **ToS** field, enter the type of service (ToS) for the FTP packets that are sent (number between 0 and 255).
iv. Select a **Mode** option: either **Active** or **Passive**.

> Passive mode is required if the firewall is enabled on your appliance.

v. Click the **Source IP** drop-down and select the IP address from which to issue the request. If a router has multiple interfaces and some interfaces do not have access to the FTP server, you must select the IP address of the interface that is to issue the request. If you leave this blank the router attempts to choose the best/closest interface which may not be the interface you want.

- If you select **HTTP**, perform the following configuration steps.
  i. In the **URL** field, enter the complete, valid URL of the HTTP server.
  ii. In the **Nameserver** field, enter the IP address of the name server.
  iii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
  iv. In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLA operation (number between 0 and 255).
  v. Click the **Operation** drop-down and select an operation.
  vi. Click the **HTTP Version** drop-down and select the HTTP version.
  vii. In the **Proxy** field, enter the IP address of the proxy.
  viii. Select the **Cache** check box to cache the IP SLA.
  ix. Click the **Source IP** drop-down and select the IP address of the source device.
- If you select **ICMP Jitter**, perform the following configuration steps.
  i. Select one of the following options.
    - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
    - Select **Specify Target Device By IP Address** to enter the IP address of the target device in a text field in the next step.
  ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
  iii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
  iv. In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLAs operation (number between 0 and 255).
  v. In the **Packet #** field, enter the number of packets to send.
  vi. In the **Interval** field, enter the number of milliseconds to be the interval between packets.
  vii. Click the **Source IP** drop-down and select the IP address of the source device.
- If you select **RTP**, perform the following configuration steps.
  i. Select one of the following options.
    - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
    - Select **Specify Target By IP** to enter the IP address of the target device in a text field in the next step.
  ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
  iii. Click the **Codec** drop-down and select the codec to use for the Mean Opinion Score (MOS), and the Impairment/calculated planning impairment factor (ICPIF) score.
  iv. In the **Source Voice Port** field, enter the voice port name, such as 0/1/1. This is not a TCP/UDP port number.
  v. In the **ICPIF Factor** field, enter the calculated planning impairment factor number that determines the type of access and how the service is to be used. (0=Conventional Wire Line, 5=Mobility Within Building, 10=Mobility Within Geographic Area, 20=Access to Hard-to Reach Location).
  vi. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
  vii. In the **Duration** field, enter the IP SLA duration.

viii. Click the **Source IP** drop-down and select the IP address of the source device.

- If you select **TCP Connect** or **UDP Echo**, perform the following configuration steps.
  i. Select one of the following options.
     - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
     - Select **Specify Target By IP Address** to enter the IP address of the target device in a text field in the next step.
  ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
  iii. In the **Target Port** field, enter the port number on which to connect (number between 0 and 65535).
  iv. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
  v. In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLAs operation (number between 0 and 255).
  vi. Click the **Source IP** drop-down and select the IP address of the source device.
- If you select **UDP Jitter**, perform the following configuration steps.
  i. Select one of the following options.
     - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
     - Select **Specify Target By IP Address** to enter the IP address of the target device in a text field in the next step.
  ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
  iii. In the **Target Port** field, enter the port number on which to connect (number between 0 and 65535).
  iv. Click the **Codec** drop-down and select the codec to use for the Mean Opinion Score (MOS) and the Impairment/calculated planning impairment factor (ICPIF) score.
  v. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
  vi. In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLAs operation (number between 0 and 255).
  vii. In the **Packet #** field, enter the number of packets to send.
  viii. In the **Interval** field, enter the number of milliseconds to be the interval between packets.
  ix. Click the **Precision** drop-down and select the precision that is dependent on the compliance revision (Revision 9 or after) of the sender and target IP SLA, you may be able to poll in microseconds.
  x. Click the **Source IP** drop-down and select the IP address of the source device.
- If you select **Video**, perform the following configuration steps.
  i. Select one of the following options.
     - Select **Specify Target By Name** to select the name of the target device from a drop-down list in the next step.
     - Select **Specify Target By IP Address** to enter the IP address of the target device in a text field in the next step.
  ii. In the **Target** field, either click the drop-down and select the target device by name or enter the target device IP address in the text field (dependent on the selection you make in the previous step).
  iii. In the **Target Port** field, enter the port number on which to connect (number between 0 and 65535).
  iv. In the **Source** field, enter the IP address of the source device.
  v. In the **Source Port** field, enter the port number of the source device.
  vi. Click the **Video Traffic Profile** drop-down.
     - Select **IPTV** to indicate the profile is for Internet Protocol television (IPTV) which is a system through which television services are delivered using the Internet Protocol Suite over a packet-switched network such as the Internet, instead of being delivered through traditional terrestrial, satellite signal, and cable television formats.
     - Select **IPVSC** to indicate the profile is for an IP surveillance camera.
     - Select **TELEPRESENCE** to indicate the profile is for the set of technologies which enable a person to feel as if they were present, or to give the appearance of being present, via telerobotics, at a place other than their true location.
  vii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
  viii. In the **Duration** field, enter the IP SLA duration.
  ix. In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLAs operation (number between 0 and 255).
- If you select **VoIP**, perform the following configuration steps.
  i. Click the **Detect Point** drop-down and select the detect point.

ii. In the **Called Number** field, enter the telephone number called.
iii. In the **Frequency** field, enter the number of seconds for how often the router should perform the test. This must be greater than 0 and should be slightly less than the poll frequency of the device.
iv. In the **ToS** field, enter the type of service (ToS) byte number in the IP header of an IP SLAs operation (number between 0 and 255).
v. In the **Source IP** field, enter the IP address of the source device.

5. Click **Save**.
6. Click  to delete a probe. This icon does not appear for probes that SevOne NMS discovers because you cannot delete those probes. When you delete a probe, the probe is removed from SevOne NMS and removed from the router. The change takes effect the next time the device is discovered. Until then, the probe displays *To Be Deleted*.

36.1.2 Import CSV

The Probe CSV Importer enables you to import probes into SevOne NMS. Use any application to create a comma delimited file. The .csv file(s) must be encoded in UTF-8. Complete all required fields. The IP SLA type entry determines what fields are required. Leave the fields that relate to other IP SLA types blank and delimited by the applicable number of commas.

File format for the required fields:

```
Probe Type, Device Name, IP SLA Type, Name, Description, Frequency, Target, Target Port, ToS, Packet #, Packet Interval, Codec Type, Probe Precision, HTTP Version, URL, Nameserver, Proxy, Cache, Detect Point, Called Number
```

Conditionally required fields for specific probe types:

```
MPID, Domain Name, Target VLAN, CoS, Target EVC, Video Traffic Profile, Source Voice Port, ICPIF Factor, Duration, Source IP, Source Port, Mode, Circuit ID, Remote ID, Subnet Mask
```

The .csv file must contain the following fields:

- **Probe Type** – (Required) Enter IP SLA
- **Device Name** - (Required) Enter the name of a device that SevOne NMS discovers, including domain name and extension.
- **IP SLA Type** - (Required) Enter one of the following: `dhcp, dlsw, dns, echo, ethernetJitter, ethernetPing, FTP, http, icmpjitter, RTP, tcpConnect, udpEcho, udpJitter, video, voip`
- **Name** - (Required) Enter the name for the probe.
- **Description** - (Required) Enter the description for the probe.
- **Frequency** - (Required) Enter is how frequently the router should perform the test, a number greater than 0.
- **Target** - (Required for DHCP, DLSw, Echo, ICMP Jitter, RTP, TCP Connect, UDP Echo, UDP Jitter, Video) - Enter either the valid name of a device in SevOne NMS or a valid IPv4 address.
- **Target Port** - (Required for TCP Connect, UDP Echo, UDP Jitter, Video) Enter a number between 0 and 65535.
- **ToS** - (Required for DLSw, Echo, FTP, HTTP, ICMP Jitter, TCP Connect, UDP Echo, UDP Jitter, Video, VoIP) Enter a number between 0 and 255. This stands for Type of Service.
- **Packet #** - (Required for Ethernet Jitter, ICMP Jitter, UDP Jitter) Enter a number.
- **Interval** - (Required for Ethernet Jitter, ICMP Jitter, UDP Jitter) Enter a number.
- **Codec Type** - (Required for RTP, UDP Jitter) Enter one of the following: `notApplicable, g711ulaw, g711alaw, g729a`
- **Precision** - (Required for UDP Jitter) Enter one of the following: `milliseconds, microseconds`
- **HTTP Version** - (Required for HTTP) Enter one of the following: `0.9, 1.0, 1.1`
- **URL** - (Required for FTP, HTTP) Enter a valid URL.
- **Nameserver** - (Required for DNS, HTTP) Enter a valid IPv4 address.
- **Proxy** - (Required for HTTP) Enter a valid IPv4 address.
- **Cache** - (Required for HTTP) Enter either Yes for enabled or No for disabled.
- **Detect Point** - (Required for VoIP) Enter one of the following: `voipDTAlertRinging, voipDTConnectOK`
- **Called Number** - (Required for VoIP) Enter a valid phone number.

The .csv file may require some or all of the following fields:

Enter commas to account for all fields up to the last field required for each specific IP SLA type. Leave non-applicable fields empty when not required for the protocol.

Example: Video must have five comma delimited empty fields (for MPID, Domain Name, Target VLAN, CoS, Target EVC) before the four required Video fields. You do not need to enter commas for any subsequent fields.

- **MPID** - (Required for Ethernet Jitter, Ethernet Ping) Enter a number.
- **Domain Name** - (Required for DNS, Ethernet Jitter, Ethernet Ping) Enter a valid domain.
- **Target VLAN** - (Required for Ethernet Jitter, Ethernet Ping) Enter a number between 1 and 4095 (inclusive).
- **CoS** - (Required for Ethernet Jitter, Ethernet Ping) Enter a number between 0 and 255.
- **Target EVC** - (Required for Ethernet Jitter, Ethernet Ping) Enter a valid number.
- **Video Traffic Profile** - (Required for Video) Enter one of the following: `IPTV, IPVSC, TELEPRESENCE`.
- **Duration** - (Required for RTP, Video) Enter a number.
- **Source IP** - (Required for DNS, Echo, FTP, HTTP, ICMP Jitter, RTP, TCP Connect, UDP Echo, UDP Jitter, Video, VoIP, Optional for DHCP) Enter a valid IPv4 address
- **Source Port** - (Required for RTP, Video) Enter a number between 0 and 65535.
- **Mode** - (Optional for FTP) Enter Active or Passive. Defaults to Passive if left blank.
- **Circuit ID** - (Optional for DHCP) Enter a hex number between 000000 and ffffff.
- **Remote ID** - (Optional for DHCP) Enter a hex number between 000000 and ffffff.
- **Subnet Mask** - (Optional for DHCP) Enter a hex number between 000000 and ffffff.

Perform the following steps to import the IP SLAs.

1. Open the .csv file, copy all the contents and paste them into the text field on the Probe CSV Importer.
2. Edit the information in the text field.
3. Click **Save**.
4. A message appears below the text field to display the success or failure of the import. If an error occurs during the import, the message describes the error and the offending lines display. SevOne NMS imports all probes that do not have errors. Probes with errors are not imported; fix the lines with errors, paste them, and repeat the steps.

36.2 Messages

The Messages tab enables you to view the messages SevOne NMS generates during the discovery of probes. These messages occur each time SevOne NMS issues an **snmpset** command to show the command and the result of any errors that occur. The following probe message data displays in the list.

- **Probe Type** – Displays the probe type (currently IP SLA is the only supported probe type).
- **Device** - Displays the name of the source device.
- **Time** - Displays the time when SevOne NMS sent the command.
- **Original Message** - Displays the probe message.
- **Success** - Displays *Yes* when the command is successful or displays *No* when the command is unsuccessful. Yes appears only for the first successful discovery of each probe.
- **Retries** - Displays the number of times the command was sent.

Select the check box for each message to manage and the following controls enable you to manage the messages.

- Click  and select **Retry Commands** to retry the probe command.
- Select **Mark Successful** to force the success to be marked as *Yes*.
- Select **Mark Unsuccessful** to force the success to be marked as *No*.
- Select **Delete Selected** to delete the probe messages you select.
- Click the **Source Device** drop-down and select a source device.
- Click **Device Summary** to display a link to the Device Summary and links to report templates that are applicable for the device.
- Click **Device Editor** to navigate to the Edit Device page for the source device.
- Click on the **Probe Type** link to display the command in a pop-up that also enables you to retry the command.

37 IP SLA

IP SLAs enable you to monitor network performance between two Cisco routers. IP SLA is a feature embedded in the Cisco IOS software that the IP SLA plugin polls to help Cisco customers understand IP service levels, increase productivity, lower operational costs, and reduce the frequency of network outages. IP SLA actively monitors network performance and helps troubleshoot your network, assess network readiness, and monitor network health. The Probe Manager enables you to manage how SevOne NMS monitors the Cisco IOS IP SLAs on the devices for which you enable the IP SLA plugin.

IP SLA technology allows remote configuration over SNMP. SevOne NMS uses this remote aspect to create IP SLAs on a router without the need to log in and run the router commands and without writing to the router startup config file. The IP SLA plugin detects and monitors all IP SLAs on the router including the IP SLAs you create.

37.1 IP SLA Identity

Unlike interfaces, IP SLAs are not physical which poses a challenge for performance monitors.

> **Example**
>
> When a device reboots there is no rule to preserve the IDs of the IP SLAs or even to preserve the IP SLA.

To avoid duplicate IP SLAs, SevOne NMS uses three criteria to determine if an IP SLA is one SevOne NMS has already encountered.

- IP SLA Type - SevOne NMS compares the IP SLA type to see if it is the same as an existing IP SLA.
- Owner - SevOne NMS compares the owner to see if it is the same as an existing IP SLA. The owner is the string that the creator of the IP SLA uses to identify itself.
- Tag - SevOne NMS compares the tag to see if it is the same as an existing IP SLA. The tag is the unique identifier that the creator uses to distinguish the IP SLA.

If the IP SLA type, owner, and tag are the same, then SevOne NMS assumes that the two IP SLAs refer to the same thing and does not create a duplicate IP SLA. Whereas the router can use the IP SLA identifier number to distinguish between IP SLAs (because the router does not need to track the IP SLAs it created before the reboot), SevOne NMS does not rely on the IP SLA numeric identifier.

37.2 IP SLA Compliance Revisions

The information on the Cisco IP SLA Compliance Revision page applies to the Cisco IP SLA MIB known as CISCO-RTTMON-MIB. The page displays a list of compliance revisions for the MIB. Cisco devices that support IP SLA also support one of these revisions. SevOne NMS checks for the existence of certain values of rttMonSupportedProtocolsValid and rttMonSupportedRttTypesValid to detect the compliance revision for a device.

37.3 Supported IP SLAs

SevOne NMS supports the following IP SLAs.

- Dhcp - This has the router perform an IP address lease request/tear down operation.
- Dlsw - This has the router perform a keep alive operation to measure the response time of a DLSw peer.
- DNS - This has the router perform a name lookup of an IP address or host name.
- Echo - This has the router perform a timed echo request/response operation.
- Ethernet Jitter - This has the router send and receive Ethernet data frames between Ethernet Connectivity Fault Management (CFM) maintenance end points (MEPs) to measure the latency, jitter, and frame loss between two MEPs.
- Ethernet Ping - This has the router send and receive Ethernet data frames between Ethernet Connectivity Fault Management (CFM) maintenance end points (MEPs) to measure the latency between two MEPs.
- FTP – This has the router download an FTP file and record how long it takes. This is a good way to determine if an important FTP site/file is available at a remote location.
- HTTP - This has the router perform a round-trip time to get a web page.
- ICMP jitter - This has the router perform delay variance analysis.
- RTP – This has the router gather network performance related statistics for a call. Available statistical measurements for VoIP networks include jitter, frame loss, Mean Opinion Score for Conversational Quality (MOS-CQ), and Mean Opinion Score for Listening Quality (MOS-LQ).
- tcpConnect - This has the router perform a timed TCP connect operation.
- udpEcho - This has the router perform a timed UDP packet send/receive operation.
- UDP Jitter - This has the router perform a delay variance analysis over UDP, usually to simulate voice traffic.
- Video - This has the router perform a one way video operation, streamed from the source to the destination.
- VoIP - This has the router measure network response time to set up a VoIP call.

37.3.1 dhcp

The IP SLA plugin collects the following dhcp data.

- Availability - Whether the IP SLA succeeded or not.
- Average Time - How long the operation took.

37.3.2 dlsw

The IP SLA plugin collects the following dlsw data.

- Availability - Whether the IP SLA succeeded or not.
- Response Time - How long the operation took.

37.3.3 DNS

The IP SLA plugin collects the following DNS data.

- Availability - Where the IP SLA succeeded or not.
- Response Time - How long the operation took.

37.3.4 echo

The IP SLA plugin collects the following echo data.

- Availability - Whether the IP SLA succeeded or not.
- Ping Time - How long the operation took.

37.3.5 Ethernet Jitter

The IP SLA plugin collects the following Ethernet Jitter data.

- Availability - Where the IP SLA succeeds or not.
- Average Delay DS - The average delay from the destination to the source.
- Average Delay SD - The average delay from the source to the destination.
- Average Jitter - The average jitter. See calculation description below.
- Average Jitter SD - The average jitter from the source to the destination.
- Average Jitter DS - The average jitter from the destination to the source.
- Average RTT - The average round trip time.
- Frames Unprocessed - The number of frames that were not processed due to high CPU load.
- Interarrival Jitter In - The mean deviation (smoothed absolute value) of the difference in frame spacing for a pair of packets from destination to source.
- Interarrival Jitter Out - The mean deviation (smoothed absolute value) of the difference in frame spacing for a pair of packets from source to destination.
- Late Frames - The number of frames that arrived late.
- Lost Frames - The number of frames that did not arrive.
- Frames Skipped - The number of frames skipped.
- Negative Jitter Average - The number of frames that reduced jitter.
- Negative Jitter Percent - The percentage of frames that reduced jitter.
- Lost Frames DS - The frame lost from the destination to the source.
- Frame Loss Ratio - The ratio of lost frames to total frames.
- Lost Frames SD - The frames lost from source to the destination.
- Frames Out of Sequence - The number of frames received out of sequence.
- Positive Jitter Average - The number of frames it takes to compensate for jitter.
- Positive Jitter Percent - The percentage of frames that introduced jitter.
- Sent Frames - The number of frames sent.

37.3.6 Ethernet Ping

The IP SLA plugin collects the following Ethernet Ping data.

- Availability - Where the IP SLA succeeded or not.
- Response Time - How long the operation took.

37.3.7 FTP

The IP SLA plugin collects the following FTP data.

- Availability - Where the IP SLA succeeded or not.
- Response Time - How long the operation took.

37.3.8 HTTP

The IP SLA plugin collects the following HTTP data.

- Availability - Whether the IP SLA succeeded or not.
- Response Time - How long the operation took.

37.3.9 ICMP jitter

The IP SLA plugin collects the following ICMP jitter data.

- Availability - Where the IP SLA succeeded or not.
- Average Delay DS - The average delay from the destination to the source.
- Average Delay SD - The average delay from the source to the destination.
- Average Jitter - The average jitter. See calculation description below.
- Average RTT - The average round-trip time. See calculation description below.
- Interarrival Jitter In - The mean deviation (smoothed absolute value) of the difference in packet spacing for a pair of packets from destination to source.
- Interarrival Jitter Out - The mean deviation (smoothed absolute value) of the difference in packet spacing for a pair of packets form source to destination.
- Late Packets - The number of packets that arrived late.
- Lost Packets - The number of packets that did not arrive.
- Negative Jitter Average - The number of packets that introduced jitter.
- Negative Jitter Percent - The percentage of packets that introduced jitter.
- Packet Loss Ratio - The ratio of lost packets to total packets.
- Packets Out of Sequence - The number of packets received out of sequence.
- Positive Jitter Average - The number of packets it takes to compensate for jitter.
- Sent Packets - The number of packets sent.

37.3.10 RTP

The IP SLA plugin collects the following RTP data.

- Availability - Where the IP SLA succeeded or not.
- Connection Time - How long the operation took.

37.3.11 tcpConnect

The IP SLA plugin collects the following tcpConnect data.

- Availability - Where the IP SLA succeeded or not.
- Connection Time - How long the operation took.

37.3.12 udpEcho

The IP SLA plugin collects the following udpEcho data.

- Availability - Where the IP SLA succeeded or not.
- Echo Time - How long the operation took.

37.3.13 UDP Jitter

The IP SLA plugin collects the following UDP Jitter data.

- Availability - Where the IP SLA succeeded or not.
- Average Delay DS - The average delay from the destination to the source.

- Average Delay SD - The average delay from the source to the destination.
- Average Jitter - The average jitter. See calculation description below.
- Average RTT - The average round-trip time. See calculation description below.
- Bandwidth - The bandwidth measure of the volume of data used for the operation over a known period of time, measured in bits per second. This uses the following calculation: the number of packets, multiplied by the request size (plus 12 bytes for protocol overhead). Multiply that by 8 (to convert from bytes to bits), and then divide by the interval between tests.
- Calculated Planning Impairment Factor (ICPIF) - Attempts to quantify, for comparison and planning purposes, the key impairments to voice quality that are encountered in the network.
- Interarrival Jitter In - The mean deviation (smoothed absolute value) of the difference in packet spacing for a pair of packets from destination to source.
- Interarrival Jitter Out - The mean deviation (smoothed absolute value) of the difference in packet spacing for a pair of packets from source to destination.
- Late Packets - The number of packets that arrived late.
- Lost Packets - The number of packets that did not arrive.
- Mean Opinion Score (MOS) - A common benchmark to determine the quality of sound produced by codes on a scale of 1 (poor quality) to 5 (excellent quality).
- Negative Jitter Average - The number of packets that reduced jitter.
- Negative Jitter Percent - The percentage of packets that reduced jitter.
- NTP State - The NTP state of the operation.
- Packet Loss DS - The packets lost from the destination to the source.
- Packet Loss Ratio - The ratio of lost packets to total packets.
- Packet Loss SD - The packets lost form the source to the destination.
- Packets Out of Sequence - The number of packets received out of sequence.
- Positive Jitter Average - The number of packets that introduced jitter.
- Positive Jitter Percent - The percentage of packets that introduced jitter.
- Sent Packets - The number of packets sent.
- Sigma Delay DS - Standard deviation of the destination-to-source delay.
- Sigma Delay SD - Standard deviation of the source-to-destination delay.
- Sigma Jitter DS - Standard deviation of the destination-to-source jitter.
- Sigma Jitter SD - Standard deviation of the source-to-destination jitter.
- Sigma RTT - Standard deviation of the round-trip time.
- UnSync RTTs - The number of probes received that were out of sync with NTP.

37.3.14 Video

The IP SLA plugin collects the following Video data.

- Availability - Where the IP SLA succeeded or not.
- Average Delay DS - The average delay from the destination to the source.
- Average Delay SD - The average delay from the source to the destination.
- Interarrival Jitter Out - The mean deviation (smoothed absolute value) of the difference in packet spacing for a pair of packets from source to destination.
- IPDV Average Jitter - The instantaneous packet delay variation.
- Late Packets - The number of packets that arrived late.
- Lost Packets - The number of packets that did not arrive.
- Negative Jitter Average - The number of packets that reduced jitter.
- Negative Jitter Percent - The percentage of packets that reduced jitter.
- NTP State - The NTP state of the operation.
- Packet Loss Ratio - The ratio of lost packets to total packets.
- Packet Loss SD - The packets lost form the source to the destination.
- Packets Out of Sequence - The number of packets received out of sequence.
- Positive Jitter Average - The number of packets that introduced jitter.
- Positive Jitter Percent - The percentage of packets that introduced jitter.
- Sent Packets - The number of packets sent.
- UnSync RTTs - The number of probes received that were out of sync with NTP.

37.3.15 VoIP

The IP SLA plugin collects the following VoIP data.

- Availability - Whether the IP SLA succeeded or not.
- Time Until Ring - How long it takes the sender to ring the receiver.

37.4 IP SLA Jitter Operation

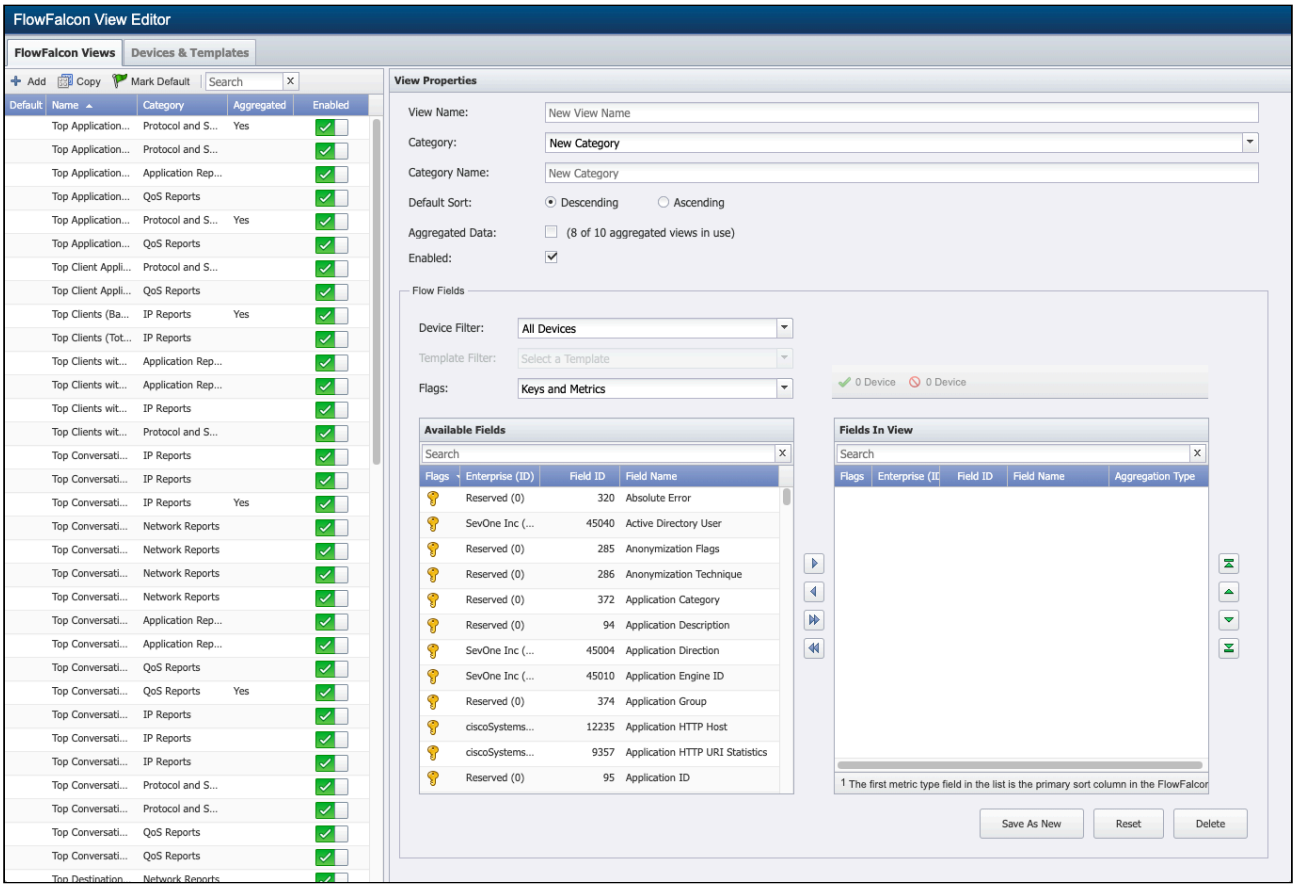
SevOne NMS calculates the following metrics for IP SLA jitter tests.

Metric	> Compliance Version 10	< Compliance Version 9
Average Round-Trip-Time (RTT)	`= rttMonLatestJitterOperRTTSum / rttMonLatestJitterOperNumOfRTT`	
Average Jitter	`= rttMonLatestJitterOperAvgJitter`	= (rttMonLatestJitterOperSumOfPositivesDS + rttMonLatestJitterOperSumOfNegativesDS + rttMonLatestJitterOperSumOfPositivesSD + rttMonLatestJitterOperSumOfNegativesSD) / (rttMonLatestJitterOperNumOfPositivesDS + rttMonLatestJitterOperNumOfNegativesDS + rttMonLatestJitterOperNumOfPositivesSD + rttMonLatestJitterOperNumOfNegativesSD)

38 FlowFalcon View Editor

The FlowFalcon View Editor enables you to define which flow template fields that devices send to SevOne NMS are used in the FlowFalcon views you use to create FlowFalcon reports. The FlowFalcon Views tab enables you to add flow template fields to the FlowFalcon views that generate FlowFalcon reports. The Devices & Templates tab displays a list of the devices you enable to send flow data to SevOne NMS plus the source template fields and option template fields that the device sends.

To access the FlowFalcon View Editor from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **FlowFalcon View Editor**. The **FlowFalcon Reports** page also provides access to the FlowFalcon View Editor.



38.1 FlowFalcon Views

The FlowFalcon Views tab enables you to manage the FlowFalcon views you use to create FlowFalcon reports. FlowFalcon views use flow data fields from the flow templates to display reports of flow statistics.

There are two types of FlowFalcon views.

- **Aggregation Disabled** views use the raw flow data to allow for more specificity in the result set at the trade off of longer report execution times and less historical data availability.
- **Aggregation Enabled** views use aggregated flow data to present the most relevant flow data for faster report creation. Your SevOne appliance hardware determines the maximum number of aggregated views (between 5 and 20). Aggregation enabled views display an asterisk in the list.

38.1.1 View List

The view list displays the following information.

- **Default** - Displays  next to the view that is used by default for quick chain reports.
- **Name** - Displays the view name.
- **Category** - Displays the category name to which the view is a member.

- **Aggregated** - Displays Yes for views that use aggregated flow data or displays nothing for views that use raw flow data.
- **Enabled** - Displays  for views that are enabled for use in reports or displays  for views that do not appear in the list of views for which you can create a report.

38.1.2 Manage FlowFalcon Views

SevOne NMS provides a starter set of FlowFalcon views to enable you to create FlowFalcon reports right out of the box and to help create FlowFalcon views that are specific to your network.

Click on a view in the list to populate the View Properties section and the Flow Fields sections on the right with the flow template fields that are available to add to the view and the flow template fields that are in the view.

-  **# Devices** - Displays the number of devices that send flow template data that could be used by the FlowFalcon view. Data from these devices could appear in a FlowFalcon report if you use this FlowFalcon view to generate the report.
-  **# Devices** - Displays the number of devices that do not send flow template data that the view supports.

Click  or  to display the Supported Devices pop-up that lists the name and IP Address of the devices that send data that the view supports and the names of the devices that do not send flow template data that appears in the view.

> If you select a view that has aggregation enabled, when you click **Save**, all aggregation data that exists for the view is deleted, even if you do not make any changes.

1. Either click **Add** above the view list or select a view in the list to manage FlowFalcon views.
2. In the **View Name** field, enter the view name.
3. Click the **Category** drop-down.
   - Select the category in which to include the view.
   - Select **New Category** and enter the category name in the **Category Name** field to add a category.
4. Select a **Default Sort** option.
   - Select **Ascending** to sort data from low value to high value.
   - Select **Descending** to sort data from high value to low value.
5. Select the **Aggregated Data** check box to create an aggregated view that uses aggregated flow data. At present, there is a limit of 10 aggregated views your appliance can support. Leave clear to create a view that uses raw flow data.

   > When you clear the check box in edit workflows, a message informs you that any aggregated data associated with the view will be deleted. Click **OK** on the message but be aware that when you click Save, all aggregated data that is associated with the view is deleted.

6. Select the **Enabled** check box to enable users to use the view in FlowFalcon reports.
7. The **Flow Fields** section enables you to select the flow template fields to include in the view. Filters enable you to limit the fields that appear in the Available Fields list.
   - Click the **Device Filter** drop-down and select the device from which to display fields.
   - Click the **Template Filter** drop-down to further filter the list to the fields in a specific template for the device you select in the previous filter.
   - Click the **Flags** drop-down and select to display only **Keys**, only **Metrics**, or both **Keys and Metrics**.
8. Move fields from the **Available Fields** list to the **Fields In View** list to include fields in the view. The fields display in the report in the sequence in which they appear in the Fields In View list and the first metric type field is the field on which the report sorts.

   > Under **Fields in View**, you can select multiple aggregation types. When you click on the **Aggregation Type**, you are presented with a drop-down list with Sum, Average, Average Non-zero, or Max options. You can choose one or more aggregation types from this drop-down list which is obtained when you click on the aggregation type of the metric already selected.

   > Under **Available Fields**, the following field names have the **Flags** column set to **Metrics** ( ) instead of **Key** (). However, if the FlowFalcon View is using the field as a **Key** then, it will not change the **Flags** column for that particular field name from Keys to Metrics.
   >
   > - TCP ACK Total
   > - TCP FIN Total
   > - TCP PSH Total

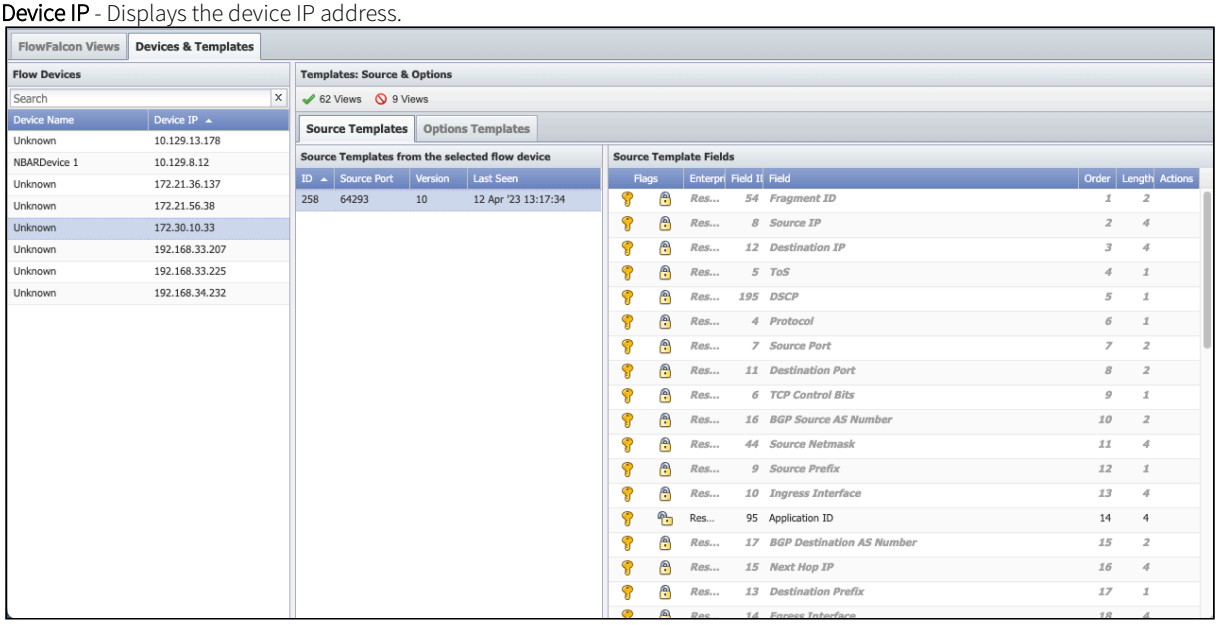
- TCP RST Total
- TCP SYN Total
- TCP URG Total

9. Click one of the following.
    - When you edit a view, click **Save** to overwrite the original view with the changes you make. This deletes any existing aggregated data for an aggregated view.
    - Click **Save as New** to create a copy of the view. This preserves aggregated data for the original view when you edit an aggregated view. (The new aggregated view starts out with no aggregated data.)
    - When you edit a view, click **Delete** to delete the view and any associated aggregated data.

38.2 Devices & Templates

The Devices & Templates tab displays the devices you enable to send flow data to SevOne NMS. When you select a device in the list, the right side displays the templates (packages) the device sends.

- **Device Name** - Displays the device name. Unknown devices are those for which you do not enable the SNMP plugin and therefore cannot have the name resolved.
- **Device IP** - Displays the device IP address.



Select a device in the list to populate the Templates: Source & Options section with the source template data and the options template data the device sends. Each device can send multiple templates.

- **# Views** - Displays the number of FlowFalcon views that support the display of data from the device. Data from this flow template could appear in a FlowFalcon report if you use any of these FlowFalcon views to generate the report.
- **# Views** - Displays the number of FlowFalcon views that do not support the display of data from the device.

Click or to display the Views Support pop-up that lists the names of the FlowFalcon views that support the flow template data and the names of the views that do not support the flow template data.

38.2.1 Source Templates and Options Templates

Flow template data varies depending upon the device. Most flow devices send source templates that contain fields from which performance metrics can be directly polled. Flow v9 and v10 send additional options template fields that are more descriptive yet contain valuable metadata on which to report.

Select a device in the Flow Devices list to display the source templates the device sends on the Source Devices tab and the options templates the device send on the Options Templates tab. The following information appears in the Templates section on both tabs.

The **Source Templates from the selected flow device** list and the **Options Templates from the selected flow device** list appears on the left side of the tab.

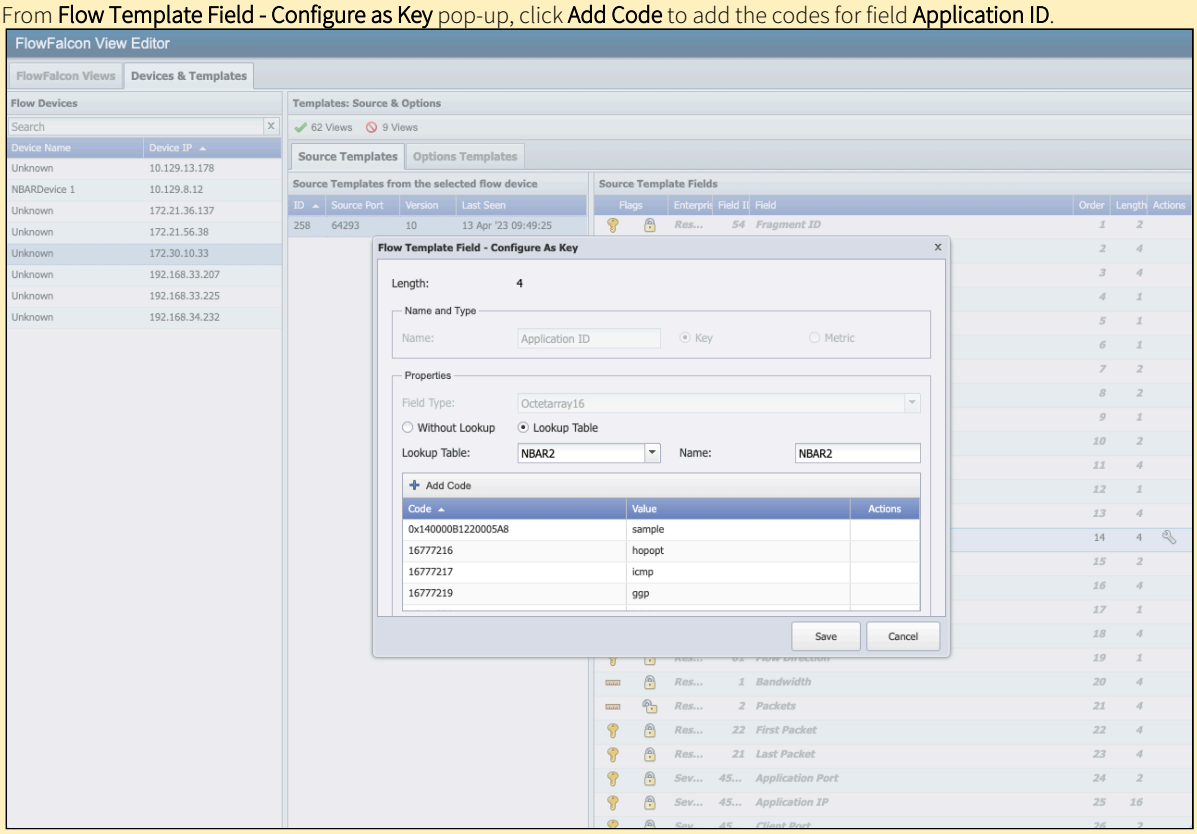
- **ID** – Displays the field identifier sent from the device with the flow template.

- **Source Port** – Displays the port on the device from which the flow template was sent.
- **Version** – Displays the flow version number.
- **Last Seen** – Displays the last time the template was received from the flow device.

Select a template in the list to display the template fields that can be used in FlowFalcon views for FlowFalcon reports.

- **Flags**:
  - - Flow field is a key.
  - - Flow field is a metric.
  - - You can edit the field.
  - - You cannot edit the field.
- **Enterprise ID** - Displays the identification of the enterprise (typically the manufacturer) that creates the field identifier.
- **Field ID** - Displays the flow template field identifier.
- **Field** - Displays the field name.
- **Order** - Displays the sequence location of the field within the flow template.
- **Length** - Displays the size of the field in bytes.

From **Devices & Templates** tab, choose a **Flow Device**. Under **Templates: Source & Options**, choose **Source Templates** tab. **Application ID** field will be available and unlocked ( ) by default. Click to configure this field. From **Flow Template Field - Configure as Key** pop-up, click **Add Code** to add the codes for field **Application ID**.



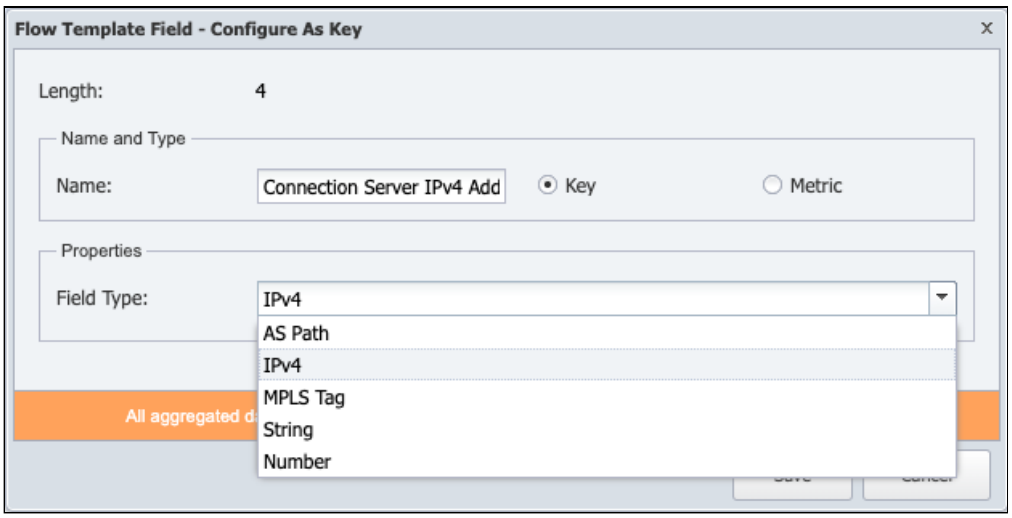
38.2.2 Edit Fields

When a field displays in the Flags column, you can perform the following steps to edit the field. This workflow varies from field to field. Steps in the following workflow appear when applicable and are disabled when they cannot be edited.

All aggregated data for every FlowFalcon view that uses the field you edit will be deleted if you save edits.

1. Click in the Actions column to display the Flow Template Field – Configure As Key/Metric pop-up.
2. In the **Name** field, edit the field name.
3. Select one of the following:
    a. Select **Key** to define the field as a key.

Example



- Click the **Field Type** drop-down. Select the appropriate field type from the drop-down. The drop-down options depend on the key length.

Key Length	Field Types
1	Direction, Protocol, String, Number
2	Port, Interface, String, Number
4	AS Path, IP, MPLS Tag, String, Number
6	MAC, String
8	String, Number, Octetarray8
16	IP Hybrid, IPv6, String, Octetarray16
32	String, Octetarray32
64	Octetarray64
128	AS Path, String, Octetarray128 ⚠ If field is a variable-length field, then **String** is the only option available. And, there is no drop-down available for this scenario.

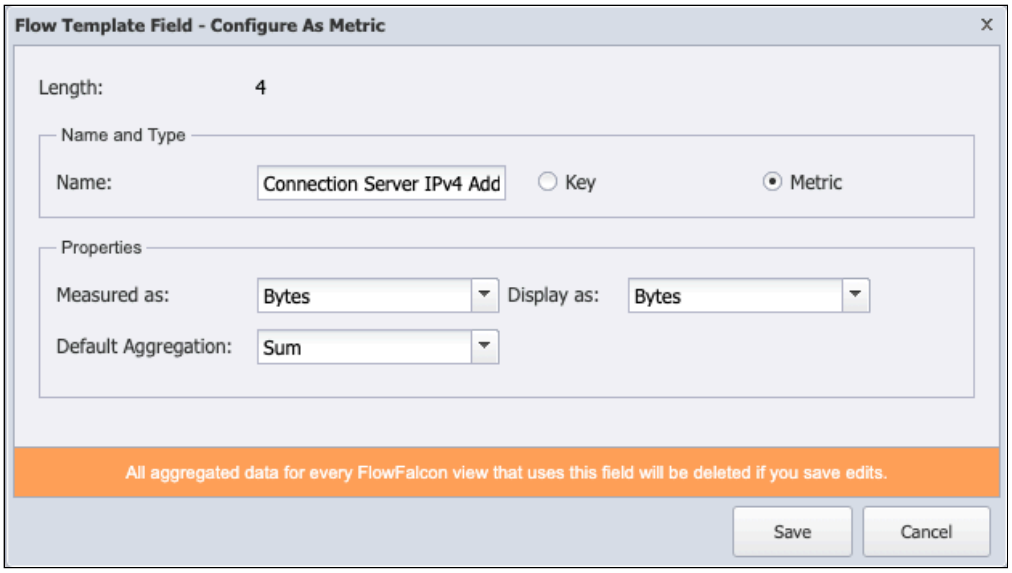
Key Length	Field Types
256	String, Octetarray256

- Select one of the following if field type **String** or **Number** is chosen.
  - Select **Without Lookup** to not use a lookup table for the field.
  - Select **Lookup Table** to use a lookup table for the field. If you select this option perform the following steps.
    a. Click the **Lookup Table** drop-down.
       - Select the lookup table for the field to use.
       - Select **New Lookup Table** and enter the lookup table name in the **Name** field to define a new lookup table.
    b. Click **Add Code** or click  to add or edit a code in the lookup table.
    c. In the **Code** field, enter the lookup table code.
    d. In the **Value** field, enter the code value.
    e. Click **Update** to save the code.
    f. Repeat to add additional codes to the lookup table.

> ⚠ If an editable field has a length of 1, 2, 4, or 8, it can also be configured as a **Metric**.

b. Select **Metric** to define the field as a metric.

Example



- Click **Measured as** drop-down to choose how to measure the metrics.
- Click **Display as** drop-down to choose how to display the metrics.
- Click the **Default Aggregation** drop-down and select the aggregation to use by default.

4. Click **Save**.

> ⚠ All aggregated data for every FlowFalcon view that uses the field you edit is deleted.

38.2.3 Synthetic Key Fields

You can combine options template fields into synthetic key fields. You create synthetic key fields on the Options Templates tab and they then appear in the list of Source Template fields on the Source Templates tab. Each options template can have multiple synthetic key field.

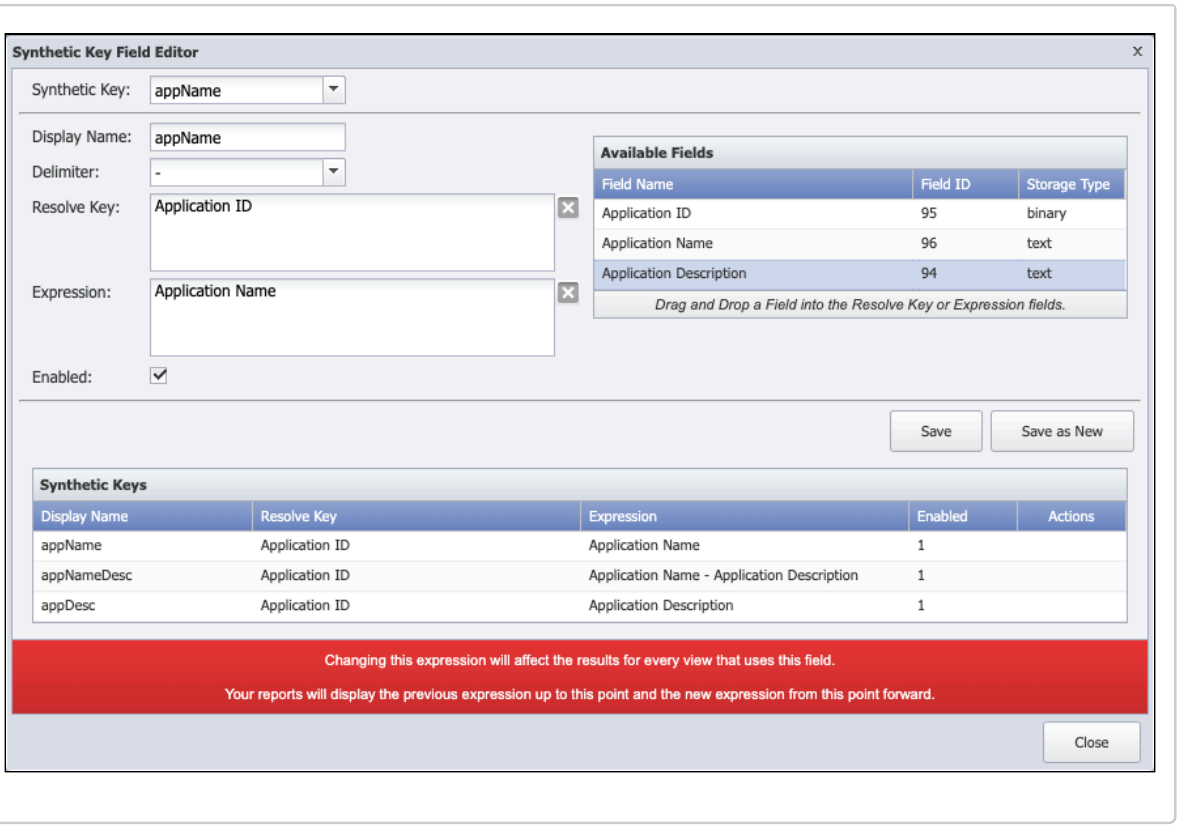
1. In the Flow Devices section, select a device to display its source templates in the Templates: Source & Options section.
2. Select the **Options Templates** tab.
3. In the **Options Templates from the selected flow device** section, select a template row to display the selected options template's fields in the Options Template Fields section.

> All fields must be configured before you can proceed to the next step. See the Edit Fields section above to configure any fields that display *Not Configured*.

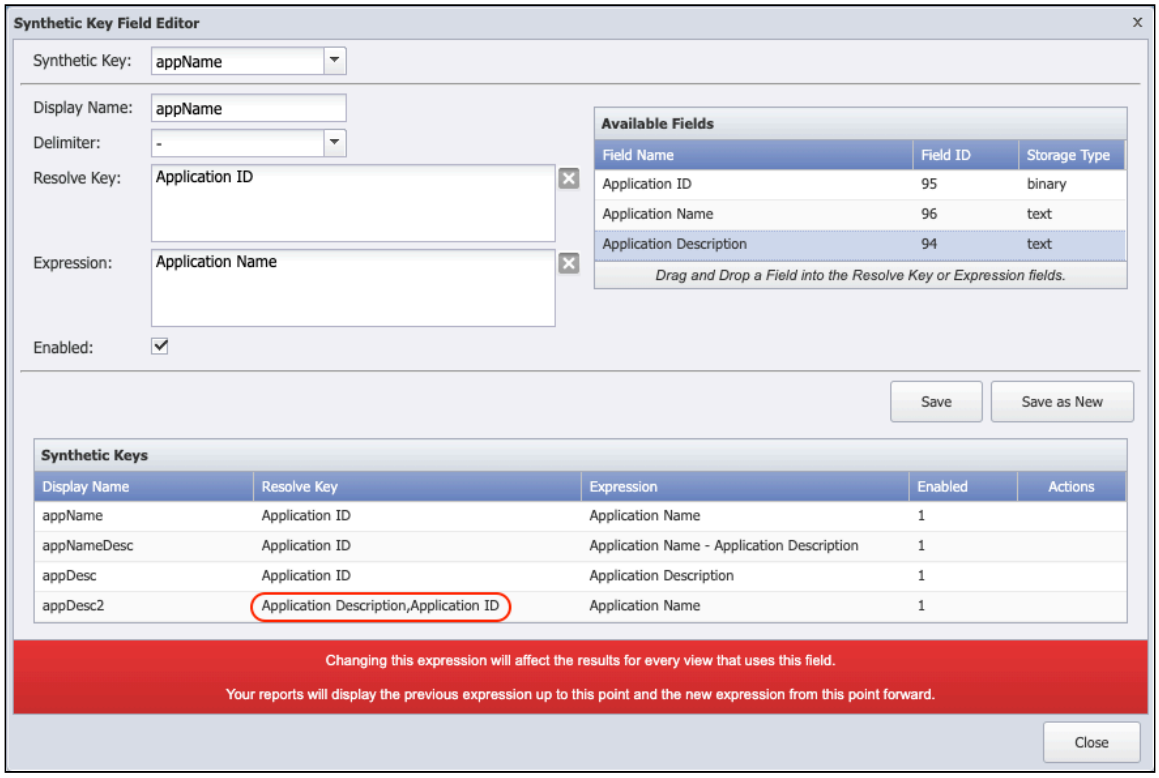
4. In the **Options Templates from the selected flow device** section Actions column, click  to display the Synthetic Key Field Editor pop-up.
5. Click the **Synthetic Key** drop-down and select an existing synthetic key from the list.

> Synthetic Key field is available only when synthetic keys exist.

6. In the **Display Name** field, enter the name to display for the field in FlowFalcon reports.
7. Click the **Delimiter** drop-down and select the delimiter to display between the fields you will add to the synthetic field.
8. Multiple synthetic keys can be created when the same Resolve Key is added one at a time. Drag a field from the **Available Fields** section into the **Resolve Key** field. The Resolve Key must be a field that exists in the source template and becomes the synthetic field into which metadata is parsed. The Resolve Key field must be a String field type.



Or, you may drag one or more fields from the **Available Fields** section into the **Resolve Key** field. The Resolve Key must be a field that exists in the source template and becomes the synthetic field into which metadata is parsed. If more than one field is added, the fields are separated by a comma. The Resolve Key field must be a String field type.



This associates the Options Templates and the Source Templates data.

9. Drag fields from the **Available Fields** section into the **Expression** field to combine the available fields into one synthetic field that displays in reports. The Expression accepts fields that have the Generic storage type and the String storage type.

   > **Resolve Key** and **Expression** fields must be different.

10. Select the **Enabled** check box to make the field available for inclusion in FlowFalcon views.
11. If you want to delete a row under Synthetic Keys, place your cursor on the row you want to delete and click  under **Actions** column.
12. To modify an existing Synthetic Key, modify the field(s) and click **Save**. This will overwrite the existing key. To save a new key, click **Save as New**.
13. When done, click **Close**.

38.3 SevOne NMS Flow Fields

SevOne NMS calculates and/or manipulates flow data to create the following fields. For fields 45050-45056, you need to understand MPLS well enough to know which MPLS attributes correspond to your network's VPN 2nd Top Layer ID, PE Egress Address, Customer VRF, Source IP Address, and Ingress PE Address. In SevOne NMS, there are three requirements to map MPLS attributes to flow data for FlowFalcon Reports.

- On the **Cluster Manager** > **Cluster Settings** tab, **FlowFalcon** subtab, select the **Enable MPLS Attribute Mapping** check box and enter the **MPLS Attribute Mapping Refresh Interval**.
- On the MPLS Flow Mapping page, upload two MPLS mapping files.
- On the **FlowFalcon View Editor**, create views that include at least one field 45040-45056.

SevOne NMS calculates and/or manipulated flow data to create the following fields.

Field #	Field Name	Field Description

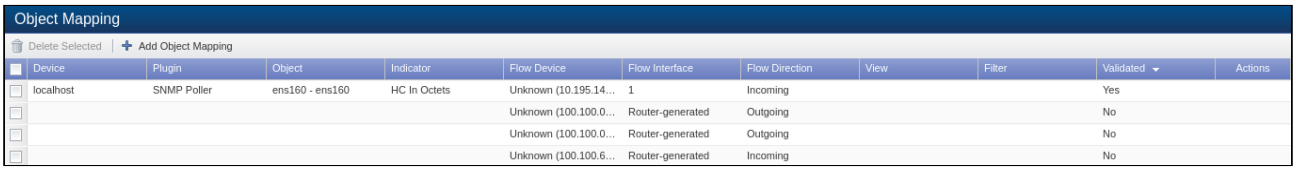
45000	Application Port	The SRC or DEST port, whichever is lower. This is the port of the application.
45001	Application IP	IPv6 address associated with the application.
45002	Client Port	Higher of SRC and DEST ports.
45003	Client IP	IPv6 address associated with the client.
45004	Application Direction	The direction of the traffic. 0 means Application Port == Source Port, 1 means Application Port == Destination Port.
45005	Next Hop IP	NetFlow view field 15 Next Hop IP is IPv4 specific and field 62 Next Hop IPv6 Address is IPv6 specific. SevOne NMS provides field 45005 Next Hop IP that pulls IPv4 from field 15 and IPv6 from field 62.
45006	Source IP Prefix	Routing prefix of the source IP address.
45007	Destination IP Prefix	Routing prefix of the destination IP address.
45010	Application Engine ID	First byte of the NBAR application id (reserved field ID 95).
45011	Application Selector ID	3 low bytes of the NBAR application id (reserved field ID 95).
45020	ToS 3-bit	First three bits of the Type of Service byte.
45021	ToS 4-bit	First four bits of the Type of Service byte.
45040	Active Directory User	The result of a look up of the client IP address in the active_directory_ips table.
45041	Peer AS	The AS of the peer for the interface through which the flow transited.
45042	Peer AS Path	The BGP path ID is the identifier SevOne NMS assigns to a route as the collector receives path updates.
45050	Customer Client IP	Customer specific IP address of the connection origin in the context of MPLS.
45051	Customer Client Subnet	Customer specific IP subnet for the connection origin in the context of MPLS.
45052	Customer VRF Name	Name of the customer VRF looked up in the database that uses MPLS_lable_2 (element 71) and PE Egress address.
45053	Customer Application IP	Customer specific IP address for connection target in the context of MPLS
45054	Customer Application Subnet	Customer specific IP subnet for connection target in the context of MPLS
45055	PE Ingress IP	IP Address of Ingress Provider Edge Router.

45056	PE Egress IP	IP Address of Egress Provider Edge Router.
45060	Service Profile	Service Profile identifier from Protocols and Services, Service Mappings, and Service Profiles.
45070	Source AS	The autonomous system number of the Source IP.
45071	Destination AS	The autonomous system number of the Destination IP.
45072	Source Country	The country code that corresponds to the Source IP.
45073	Destination Country	The country code that corresponds to the Destination IP.

39 Map Flow Objects

The Object Mapping page enables you to map the indicators on the objects that plugins poll to a flow interface. This enables you to display a FlowFalcon report of the flow data that is related to the poll data from an indicator that appears in an Instant Graph. When multiple objects rely on flow data from a single interface you can map multiple objects to a single flow interface, even if the objects are on different devices. Objects for which you define a mapping display a NetFlow button when you create an Instant Graph or enable the ability to chain from a report attachment that contains the object to a FlowFalcon attachment. The object mapping includes the designation of the FlowFalcon view and the definition of the filters for the FlowFalcon Reports page to use to display the flow information you specify. When you click the NetFlow button on an instant graph, the FlowFalcon Reports page appears with the view, settings, and filters you define from the Object Mapping page.

To access the Object Mapping page from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **Object Mapping**.



When you enable the SNMP plugin for a device and you enable the device to send flow data to SevOne NMS, most SNMP objects are automatically mapped to their corresponding flow interface. For other plugin objects and SNMP objects such as QoS, the Object Mapping page enables you to map the indicator to an interface and to define the FlowFalcon report parameters that are applicable for the data.

> **Example:** To display the flows for QoS Queues, create an object mapping that uses a FlowFalcon view that contains DSCP and has an appropriate filter to display a FlowFalcon report of the traffic that moves across the queue.

Users have access to view devices to which user has permissions. To give a user permissions to view a flow device, the user should be granted Device View access to a plugin device that is mapped to this flow device via object mapping relation.

39.1 Map List

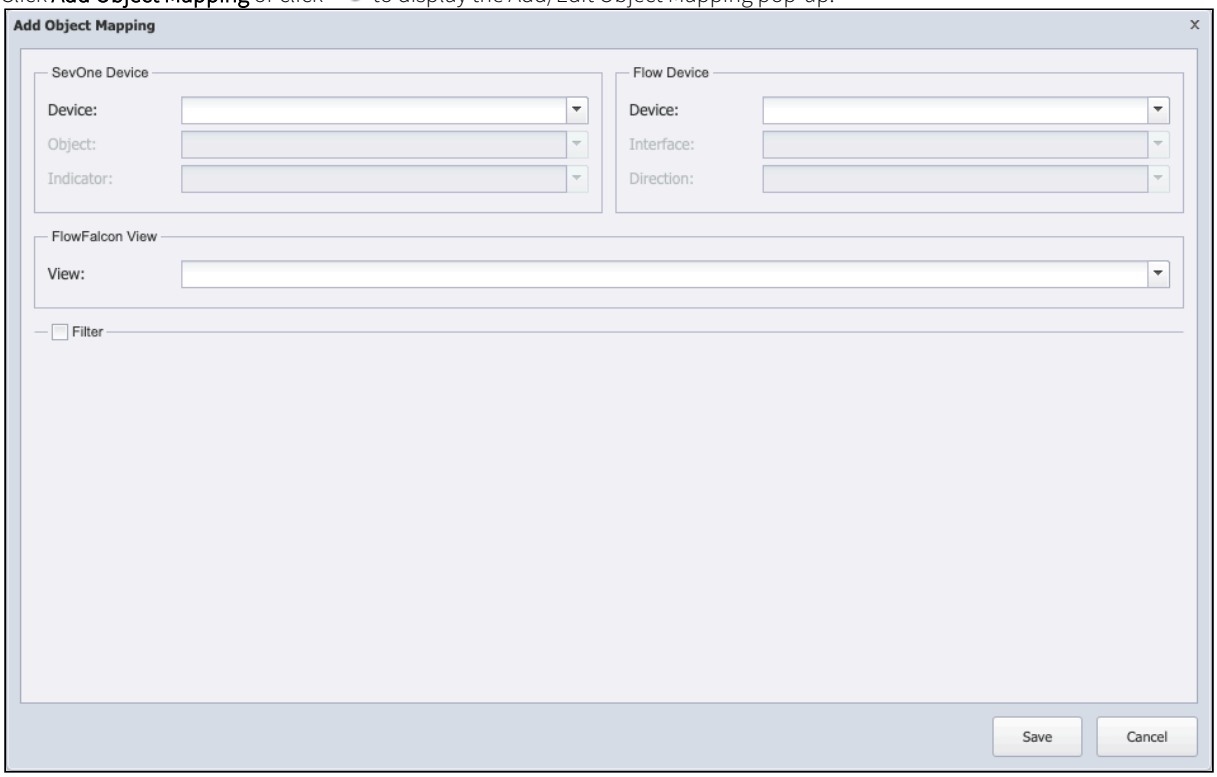
The list displays the mapping relationship between an indicator and its corresponding flow interface.

- **Device** - Displays the name of the device that contains the indicator the plugin polls.
- **Plugin** - Displays the name of the plugin that polls the object that contains the indicator.
- **Object** - Displays the name of the object that contains the indicator you map to a flow interface.
- **Indicator** - Displays the name of the indicator you map to a flow interface.
- **Flow Device** - Displays the name of the device that contains the interface that sends flow data to SevOne NMS.
- **Flow Interface** - Displays the name of the flow interface to which you map the indicator.
- **Flow Direction** - Displays the mapping direction.
- **View** - Displays the name of the FlowFalcon view the FlowFalcon Reports page uses for the FlowFalcon report that displays the flow information for the indicator/interface.
- **Filter** - Displays the name of the filter you define to apply to the FlowFalcon report.
- **Validated** - Displays all valid and invalid NetFlow entries. Valid entries are displayed as **Yes** and invalid entries are displayed as **No** in this column. When you hover over a row in this column with an invalid entry, it provides a tooltip with the exact validation failure message.

39.2 Manage Mappings

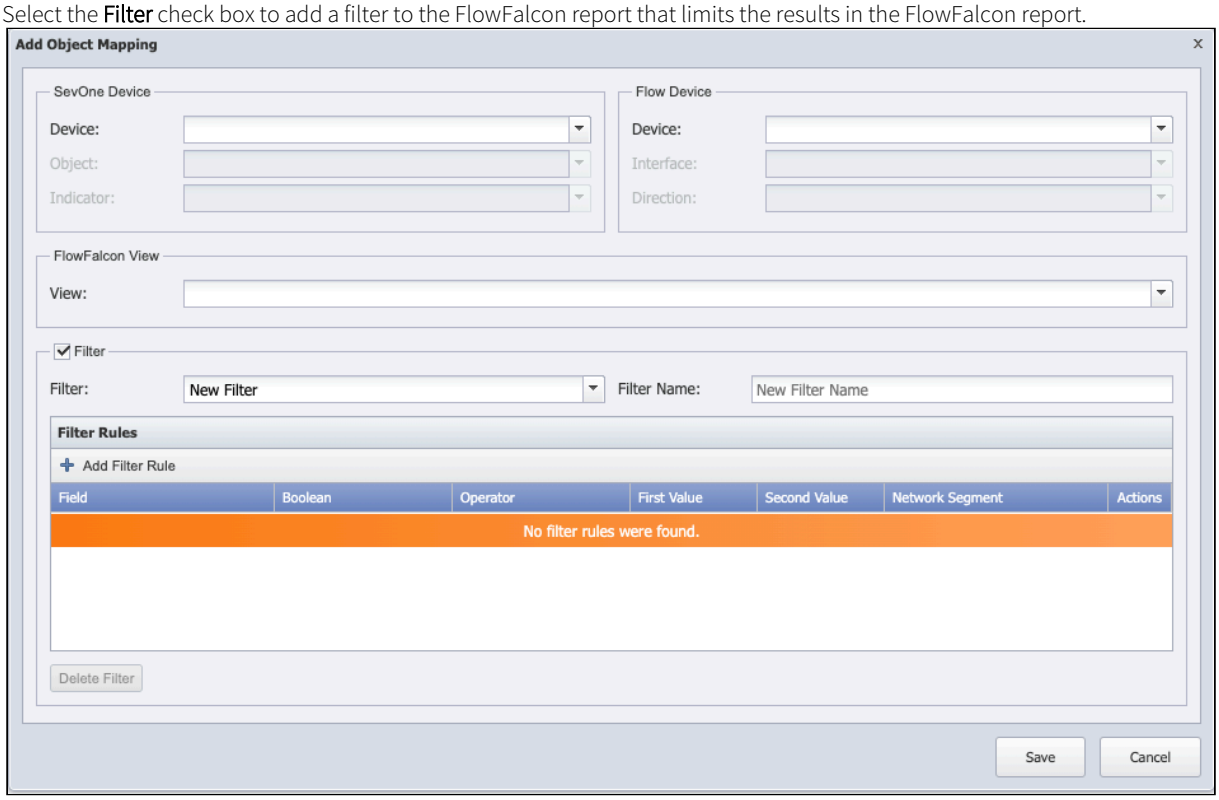
The Add/Edit Object Mapping pop-up contains four sections. The SevOne Device section enables you to select the indicator to which to map the flow interface. The Flow Device section enables you to select the flow interface to which to map the indicator. The View section enables you to select the FlowFalcon view to use for the FlowFalcon report. The Filter section enables you to define the filter to apply to the FlowFalcon report. Each filter is composed of rules and if a rule is not applicable for the FlowFalcon view, the filter ignores the rule. This enables you to create filters that are applicable for multiple FlowFalcon views.

1. Click **Add Object Mapping** or click  to display the Add/Edit Object Mapping pop-up.



2. **SevOne Device**
    a. lick the **Device** drop-down and select the device that contains the indicator.
    b. Click the **Object** drop-down and select the object that contains the indicator to map.
    c. Click the **Indicator** drop-down and select the indicator.
3. **Flow Device**
    a. Click the **Device** drop-down and select the flow device that contains the interface.
    b. Click the **Interface** drop-down and select the interface to which to map the indicator.
    c. Click the **Direction** drop-down and select the direction of the interface mapping.
4. **FlowFalcon View**
    a. Click the **View** drop-down and select the view for the FlowFalcon Reports page to display for the indicator's interface data.

5. Select the **Filter** check box to add a filter to the FlowFalcon report that limits the results in the FlowFalcon report.



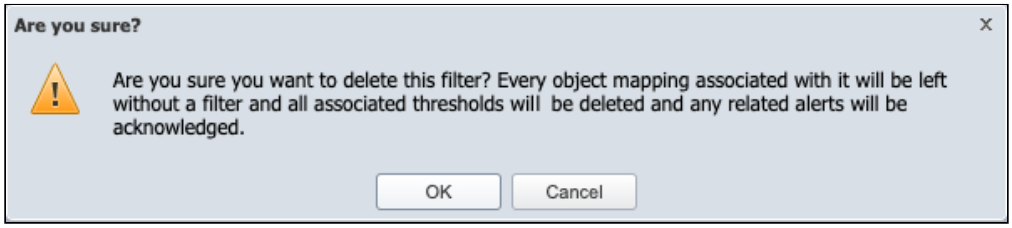
   a. Click the **Filter** drop-down and select a filter. To create a new filter, select **New Filter**.
   b. In the **Filter Name** field, enter the filter name.
   c. Click **Add Filter Rule** or click  to display a new line in the list or to make the rule editable.
      - Click the **Field** drop-down and select the field on which to apply the rule. Fields from the view you select appear first in the list followed by every known field from flow data. The FlowFalcon View Editor displays field details.
      - Click the **Boolean** drop-down and select **Is** to define the rule with an Is Boolean operator or select **Is Not** to define the rule with an Is Not Boolean operator.

        > The filter Boolean expression works such that for each **unique** field, SevOne NMS creates a Boolean expression that consists of the negative rules and the positive rules. The negative rules are AND'd to form a sub-expression and the positive rules are OR'd to form a sub-expression. These sub-expressions are then AND'd to form the final expression for each unique field. Then, each unique field's composite expression is AND'd to other field expressions.

      - Click the **Operator** drop-down and select a comparison operator. Most operators are self-evident.

        > Depending on the field you selected for **Field**, the **Operator** options may include **Mask** and **Subnet** (in addition to **Equal To**, **Greater Than**, **Less Than**, and **Between**). Select **Mask** to report on flow data that needs to match in the manner of IP address subnet masking. Select **Subnet** to report on flow data that needs to be from the subnet you select from the **Network Segment** drop-down. The **Network Segment** drop-down becomes available when you select **Subnet** as the operator. For information about defining network segments, please see the section Network Segment Manager.

      - In the **First Value** field enter the first value on which to filter data.
      - In the **Second Value** field, enter the second value in a value range, when applicable.
      - Click **Update** to save the rule.
   d. Repeat these steps to add additional rules to the filter.
   e. To delete a filter, click the **Filter** drop-down and select the filter to delete. You will get the following message. Click **Yes** to continue with the deletion; every object mapping associated with it will be without this filter, all associated policies and thresholds will be deleted, and any related alerts will be acknowledged. Click **No** to cancel.

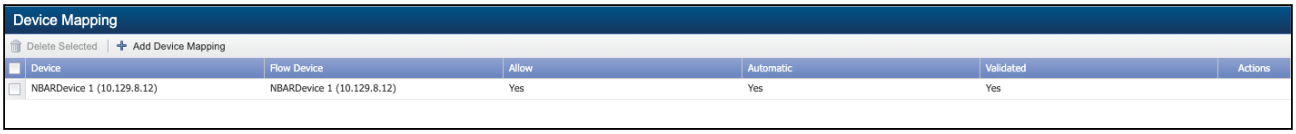


6. Click **Save** to save the object mapping changes.

40 Map Flow Devices

The Device Mapping page enables you to map a SevOne device to a flow device.

To access the Device Mapping page from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **Device Mapping**.



Users have access to view devices to which the user has permissions. To give a user permissions to view a flow device, the user should be granted Device View access to a SevOne device that is mapped to this flow device via device mapping relation.

40.1 Map List

The list displays the mapping relationship between a device and its corresponding flow device.

- **Device** - Displays the name of the SevOne device.
- **Flow Device** - Displays the name of the device that contains the interface that sends flow data to SevOne NMS.
- **Allow** - Displays if the map between SevOne device and flow device is possible. If it is set to **No** then, no mapping is going to be applied by SevOne discovery. You may edit this field by **Add/Edit Device Mapping** pop-up.
- **Automatic** - Displays if the mapping is done manually or by SevOne discovery. This field is not editable.
- **Validated** - Displays all valid and invalid NetFlow entries. Valid entries are displayed as **Yes** and invalid entries are displayed as **No** in this column. When you hover over a row in this column with an invalid entry, it provides a tooltip with the exact validation failure message.

40.2 Manage Mappings

The Add/Edit Device Mapping pop-up contains three sections. The SevOne Device section enables you to select the SevOne device to which to map the flow device. The Flow Device section enables you to select the flow device. The Allow mapping section allows you to manually map the SevOne device with the flow device.

> Leave **Allow Mapping** unchecked if the SevOne device should not map to the flow device even if they have the same IP address in SevOne NMS.

1. Click **Add Device Mapping** or click  to display the Add/Edit Device Mapping pop-up
2. In the SevOne Device section, click the **Device** drop-down and select the SevOne device.
3. In the Flow Device section, click the **Device** drop-down and select the flow device.
4. Select the **Allow Mapping** check box to manually map the SevOne device with the flow device.
5. Click **Save** to save the mapping changes.

41 FlowFalcon Views

FlowFalcon views enable you to use the flow data that devices send to SevOne NMS in FlowFalcon reports. When you enable a device to send flow data to SevOne NMS, the device sends flow packets in the format of flow templates that contain metrics and keys. The **FlowFalcon View Editor** enables you to define FlowFalcon views that are specific to your flow report requirements. For details, please refer to sections **Enable Flow Technologies** and **FlowFalcon View Editor** in *SevOne NMS System Administration Guide*.

There are two types of FlowFalcon report views.

- **Aggregation Disabled** views use raw flow data to allow for more specificity in the result set at the trade off of longer report execution times and less historical data availability.
- **Aggregation Enabled** views use aggregated flow data to present the most relevant flow data for faster report creation. You can choose to run each aggregation enabled view in the aggregation disabled mode to use raw flow data. Your SevOne appliance hardware determines the maximum number of aggregated views.

> When you create a TopN flow report (e.g., Top Talkers) based on aggregated data, the report will not be entirely precise. You can increase the value for the **Aggregation TopN** setting (go to **Cluster Manager** -> **Cluster Settings** -> **FlowFalcon**) for greater precision. However, any value greater than 100 will increase the system load, which may eventually lead to data loss.

SevOne NMS provides starter set FlowFalcon Reports views to enable you to run common flow reports right out of the box. The default view is Top Talkers (Bandwidth, Packets, Flows).

FlowFalcon View Name	Included Flow Template Fields
AGGREGATION DISABLED **Application Reports**	
Top Applications (Total Delay, Application Delay, Network Delay)	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application Port
Top Clients with Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Client IP, Protocol, Application Port
Top Clients with Client Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Client IP, Protocol, Client Port
Top Conversations with Application	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application IP, Client IP, Application Port
Top Conversations with Application and Direction	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application IP, Application Direction, Client IP, Application Port
Top Flows	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Client IP, Protocol, Application Port, Client Port
Top Flows and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Protocol, Application Port, Client Port
Top Flows with Next Hop	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Client IP, Next Hop IP, Protocol, Application Port, Client Port

FlowFalcon View Name	Included Flow Template Fields
Top Flows with Next Hop and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Next Hop IP, Protocol, Application Port, Client Port
Top Next Hops with Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Next Hop IP, Protocol, Application Port
Top Next Hops with Client Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Next Hop IP, Protocol, Client Port
Top Talkers with Application	Application IP, Application Port, Bandwidth, Total Delay (avg), Application Delay (avg), Packets
Top Talkers with Protocol and Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Protocol, Application Port
Top Talkers with Protocol and Client Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Protocol, Client Port
IP Reports	
Top Clients (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Client IP
Top Clients (Total Delay, Application Delay, Network Delay)	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Client IP
Top Clients with Next Hop	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Client IP, Next Hop IP
Top Conversations (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Client IP
Top Conversations (Total Delay, Application Delay, Network Delay)	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application IP, Client IP
Top Conversations and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP
Top Conversations with Direction	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application IP, Application Direction, Client IP
Top Conversations with Next Hop	Bandwidth, Packets, Flows, Multicast Packets, Multicast, Bandwidth, Application IP, Client IP, Next Hop IP
Top Conversations with Next Hop and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast, Bandwidth, Application IP, Application Direction, Client IP, Next Hop IP

FlowFalcon View Name	Included Flow Template Fields
Top Next Hops	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Next Hop IP
Top Talkers (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP
Top Talkers (Total Delay, Application Delay, Network Delay)	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application Port
Top Talkers with Applications	Total Delay, Application Delay, Network Delay, Bandwidth, Packets, Application IP, Application Port
Top Talkers with Next Hop	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Next Hop IP
Medianet	
Top Media Destinations	Bandwidth, Packets, Packet Loss, Interarrival Jitter, Round Trip Time, Destination IP, Destination Port
Top Media Flows	Bandwidth, Packets, Packet Loss, Interarrival Jitter, Round Trip Time, Source IP, Source Port, SSRC, Destination IP, Destination Port, DSCP
Top Media Sources	Bandwidth, Packets, Packet Loss, Interarrival Jitter, Round Trip Time, Source IP, Source Port
Network Reports	
Top Conversations AS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, BGP Source AS Number, BGP Destination AS Number
Top Conversations AS *(enriched)*	Source AS, Destination AS, Bandwidth, Packets
Top Conversations AS and Country	Source AS, Source Country, Destination AS, Destination Country, Bandwidth, Packets
Top Conversations Country	Source Country, Destination Country, Bandwidth, Packets
Top Destination AS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, BGP Destination AS Number
Top Destination AS *(enriched)*	Destination AS, Bandwidth, Packets
Top Destination Countries	Destination Country, Bandwidth, Packets
Top Destination Mask	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Destination Prefix

FlowFalcon View Name	Included Flow Template Fields
Top Source AS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, BGP Source AS Number
Top Source AS *(enriched)*	Source AS, Bandwidth, Packets
Top Source Countries	Source Country, Bandwidth, Packets
Top Source Mask	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Source Prefix
Protocol and Service Reports	
Top Applications (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application Port
Top Applications (Bi-directional)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Protocol, Application Port, Client Port
Top Applications with Protocol	Flows, Bandwidth, Packets, Multicast Packets, Multicast Bandwidth, Application Port, Protocol
Top Client Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Protocol, Client Port
Top Clients with Service	Client IP, Service Profile, Bandwidth, Packets
Top Conversations with Service	Application IP, Client IP, Service Profile, Bandwidth, Packets
Top Conversations with Service and Direction	Application IP, Application Direction, Client IP, Service Profile, Bandwidth, Packets
Top Protocols	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Protocol
Top Services (Bandwidth, Packets, Flows)	Service Profile, Bandwidth, Packets
Top Services with Protocol (Bandwidth, Packets, Flows)	Service Profile, Protocol, Bandwidth, Packets
Top Talkers with Service (Bandwidth, Packets, Flows)	Application IP, Service Profile, Bandwidth, Packets
QoS Reports	
Top Applications with Next Hop and ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application Port, Next Hop IP, ToS

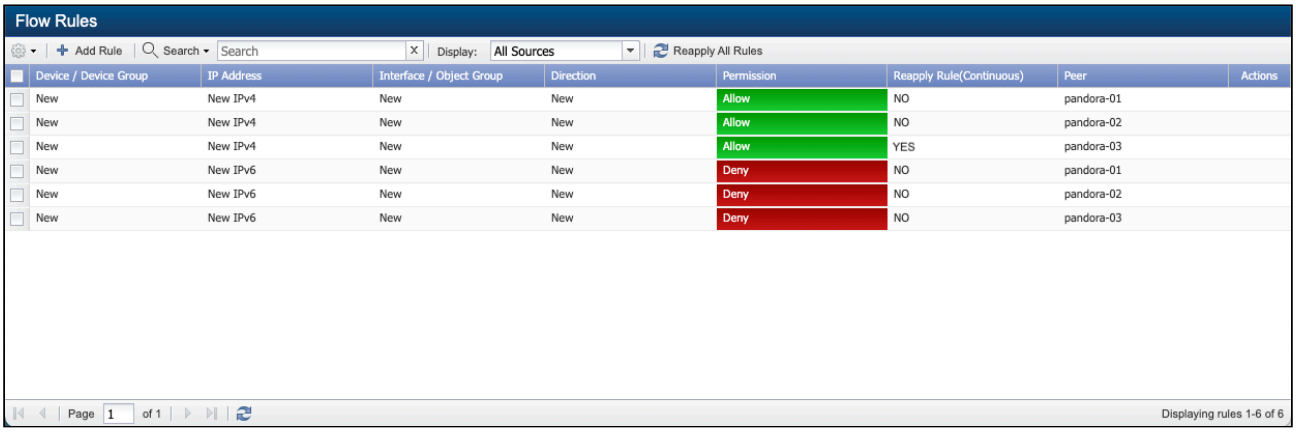
FlowFalcon View Name	Included Flow Template Fields
Top Applications with ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Protocol, Application Port, ToS
Top Client Applications with ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Protocol, Client Port, ToS
Top Conversations with Application and ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Client IP, Application Port, ToS
Top Conversations with Application and ToS and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Application Port, ToS
Top Conversations with Service and ToS	Application IP, Client IP, Service Profile, ToS, Bandwidth, Packets
Top Conversations with Service and ToS and Direction	Application IP, Application Direction, Client IP, Service Profile, ToS, Bandwidth, Packets
Top Flows with Next Hop and ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Client IP, Next Hop IP, Protocol, Application Port, Client Port, ToS
Top Flows with Next Hop and ToS and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Next Hop IP, Protocol, Application Port, Client Port, ToS
Top Flows with ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Client IP, Protocol, Application Port, Client Port, ToS
Top Flows with ToS and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Protocol, Application Port, Client Port, ToS
Top Services with Next Hop and ToS	Service Profile, Next Hop IP-1, ToS, Bandwidth, Packets
Top Services with ToS	Service Profile, ToS, Bandwidth, Packets
Top Talkers with Application and ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Protocol, Application Port, ToS
Top Talkers with Client Application and ToS	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Protocol, Client Port, ToS
Top Talkers with Service and ToS	Application IP, Service Profile, ToS, Bandwidth, Packets
Top Types of Service	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, ToS
AGGREGATION ENABLED	

FlowFalcon View Name	Included Flow Template Fields
IP Reports	
Top Clients (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Client IP
Top Conversations and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP
Top Talkers (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP
Top Talkers with Applications	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Port
Protocol and Service Reports	
Top Applications (Bandwidth, Packets, Flows)	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application Port
Top Applications with Protocol	Flows, Bandwidth, Packets, Multicast Packets, Multicast Bandwidth, Application Port, Protocol
QoS Reports	
Top Conversations with Application and ToS and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Application Port, ToS
Top Flows with Next Hop and ToS and Direction	Bandwidth, Packets, Flows, Multicast Packets, Multicast Bandwidth, Application IP, Application Direction, Client IP, Next Hop IP, Protocol, Application Port, Client Port, ToS

42 Flow Rules

The Flow Rules page enables you to define global rules to not process the flow data SevOne NMS receives. SevOne NMS evaluates the rules you define and applies the rule that is most specific to each source. When you enable devices to send flow data to SevOne NMS, SevOne NMS allows and processes all flow data by default. Networks have the potential to send large amounts of flow traffic. The Flow Rules page enables you to define global rules to deny the processing of flows. You can override the rules you define here for specific interfaces from the Flow Interface Manager.

To access the Flow Rules page from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **Flow Rules**.



42.1 Flow Rules List

The list displays all flow rules by default. Click the **Display** drop-down to display rules for **All Sources**, **Allowed Sources**, or **Denied Sources**.

- **Device / Device Group** - Displays the name of the device / device group for which the rule is applicable. Displays *New* when the rule applies to new devices / device groups that have yet to send flow to SevOne NMS. Displays *Unknown* when SNMP plugin is not enabled and the device / device group name is not resolvable.
- **IP Address** - Displays the device IP address. If resource type selected is Device Group, this field is empty.
- **Interface / Object Group** - Displays the interface or the object group for which the rule is applicable. Displays *New* when the rule applies to new interfaces / object groups that have yet to send flows to SevOne NMS.
- **Direction** - Displays *Incoming* when the rule applies to incoming traffic. Displays *Outgoing* when the rule applies to outgoing traffic. Displays *New* when the rule applies to flows that are from devices / device groups that are new in SevOne NMS.
- **Permission** - Displays *Allow* when SevOne NMS processes the flow data across the interface. Displays *Deny* when SevOne NMS does not process the flow data across the interface.
- **Reapply Rule(Continuous)** - select the check box to apply updated flow rules to existing flow interfaces that have already been discovered. This allows flow rules and flow interface manager policies to remain consistent.

  > ⚠ Object Group based rules without *reapply rule continuously* may not work as you expect because such a rule is only automatically applied for a new interface as it is first seen by the system, at which point by definition there is no object mapping for it.
  > A workaround for this is to apply the rules later by clicking the **Reapply All Rules** button.

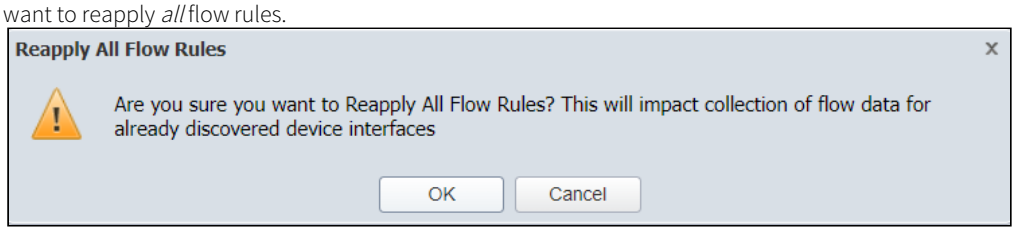
- **Peer** – Displays the name of the peer to which you define the device / device group to send flow data.

42.2 Manage Flow Rules

Perform the following steps to manage the rules in the list.

- – Select the check box for each rule to manage, click , and then select one of the following options.
  - Select **Allow Selected Flows** to process the flow data across the interface.
  - Select **Deny Selected Flows** to not process the flow data across the interface.
  - Select **Delete Selected Rules** to delete the rules.

- Click button **Reapply All Rules** for *all* flow rules to be applied. You will get the following pop-up to confirm if you are sure you want to reapply *all* flow rules.

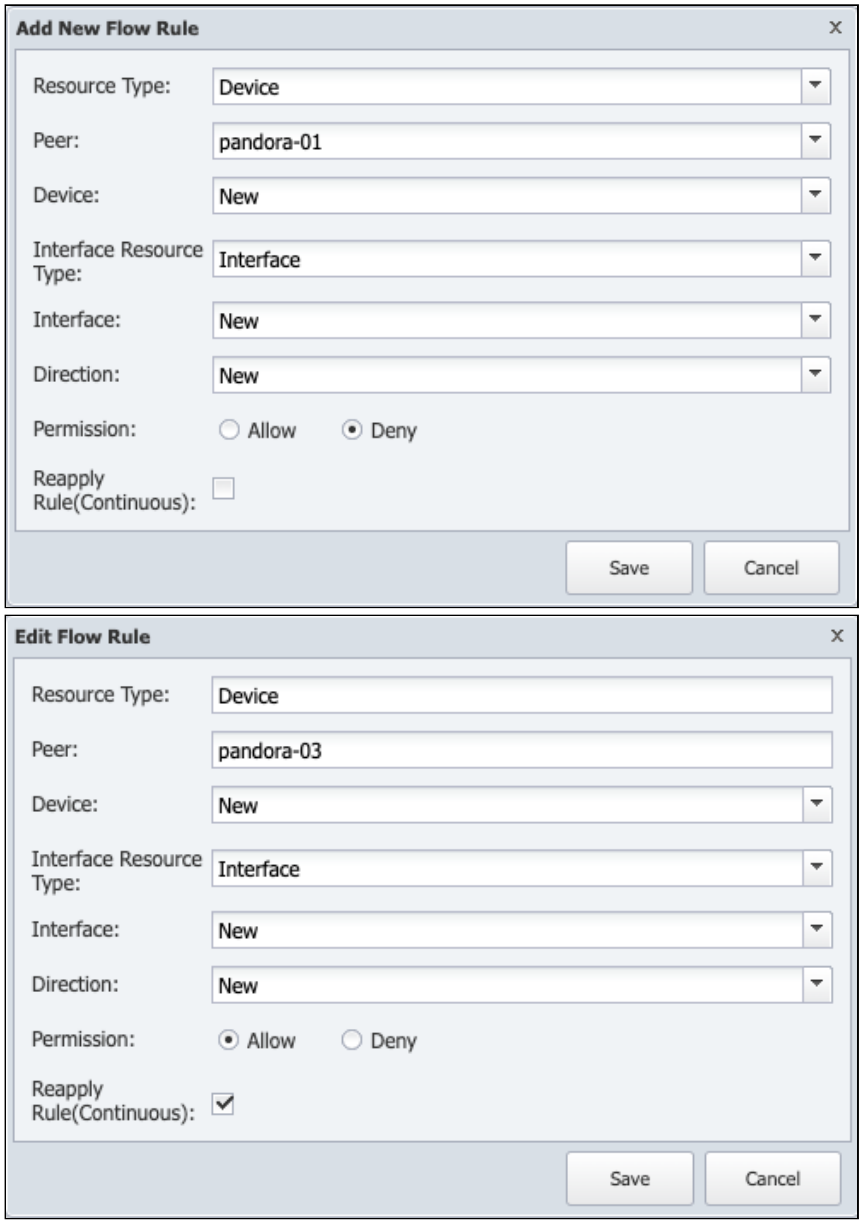


> **CAUTION**
>
> When you reapply all flow rules by clicking **OK** in the pop-up, it will impact the collection of flow data for the device interfaces that have already been discovered. **Please proceed with caution!**

Perform the following steps to add and edit flow rules.

1. Click **Add Rule** to display **Add New Flow Rule** pop-up. Or, click on the row of an existing flow rule to display **Edit Flow Rule** pop-up.

Example: Add new flow rule / Edit existing flow rule



2. Click the **Resource Type** drop-down to select **Device** or **Device Group**.
    a. **Device**
        - **Peer** - click the drop-down and select the peer to define the device to send flow data.
        - **Device** - click the drop-down and select the device from which you want to define a flow rule. If you select **Specify...**, enter the IP address in the **IP Address** field.
        - **Interface Resource Type** - click the drop-down and select **Interface** or **Object Group**.
            - if *Interface Resource Type = Interface*, then in field **Interface**, click the drop-down and select the interface for which you want to apply the rule. If you select **Specify...**, enter the interface number in the **Interface Number** field.
            - if *Interface Resource Type = Object Group*, then in field **Object Group**, click the drop-down and select an object group from the list available for which you want to apply the rule.
        - **Direction** - click the drop-down.
            - Select **New** to apply the rule to any applicable new flow incoming or outgoing.
            - Select **Incoming** to apply the rule to data that comes into the device. V5 NetFlow is an ingress technology that can only report on data that the interface receives.
            - Select **Outgoing** to apply the rule to data that goes out from the interface. For v5 NetFlow, SevOne NMS uses data from other flows to create an estimation of outgoing flows.

- **Permission** - select option Allow or Deny.
  - Select **Allow** to process the flow data across the interface.
  - Select **Deny** to not process the flow data across the interface.

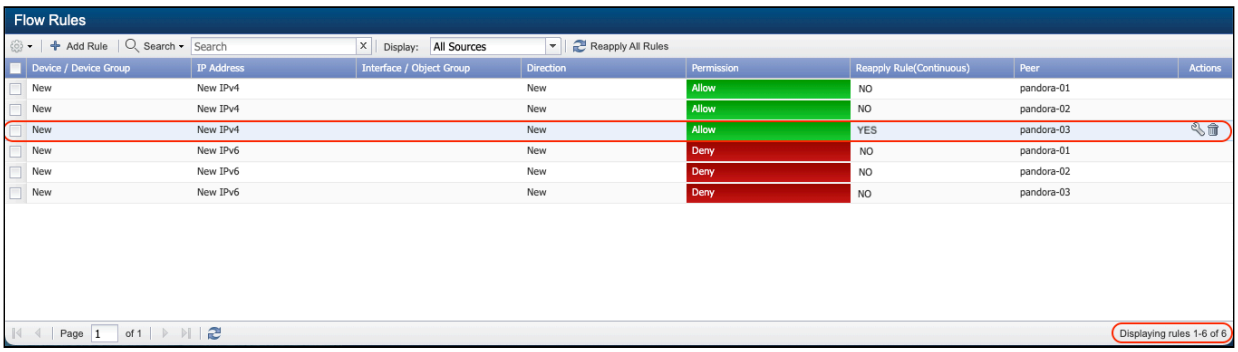
  > Click **Save**. When the rule specifies that,
  > - the device IP address is not the default IPv4 or IPv6 address,
  > - direction is not new (i.e., must be incoming / outgoing), and
  > - the interface is not new
  >
  > the rule appears in Flow Interface Manager page and not the Flow Rules page.

- **Reapply Rule(Continuous)** - select the check box to apply updated flow rules to existing flow interfaces that have already been discovered. This allows flow rules and flow interface manager policies to remain consistent.

b. **Device Group**
- **Peer** - click the drop-down and select the peer to which you define the device group to send flow data.
- **Device Group** - click the drop-down and select the device group from which you want to define a flow rule.
- **Interface Resource Type** - click the drop-down and select **Interface** or **Object Group**.
  - if *Interface Resource Type = Interface*, then field **Interface** is set to **New** by default and it means *all interfaces*. This field cannot be modified.
  - if *Interface Resource Type = Object Group*, then in field **Object Group**, click the drop-down and select an object group from the list available for which you want to apply the rule.
- **Direction** - click the drop-down.
  - Select **New** to apply the rule to any applicable new flow incoming or outgoing.
  - Select **Incoming** to apply the rule to data that comes into the device group. V5 NetFlow is an ingress technology that can only report on data that the interface receives.
  - Select **Outgoing** to apply the rule to data that goes out from the interface. For v5 NetFlow, SevOne NMS uses data from other flows to create an estimation of outgoing flows.
- **Permission** - select option Allow or Deny.
  - Select **Allow** to process the flow data across the interface.
  - Select **Deny** to not process the flow data across the interface.
- **Reapply Rule(Continuous)** - select the check box to apply updated flow rules to existing flow interfaces that have already been discovered. This allows flow rules and flow interface manager policies to remain consistent.

> Click **Save**. Rule is always added to the Flow Rules page.

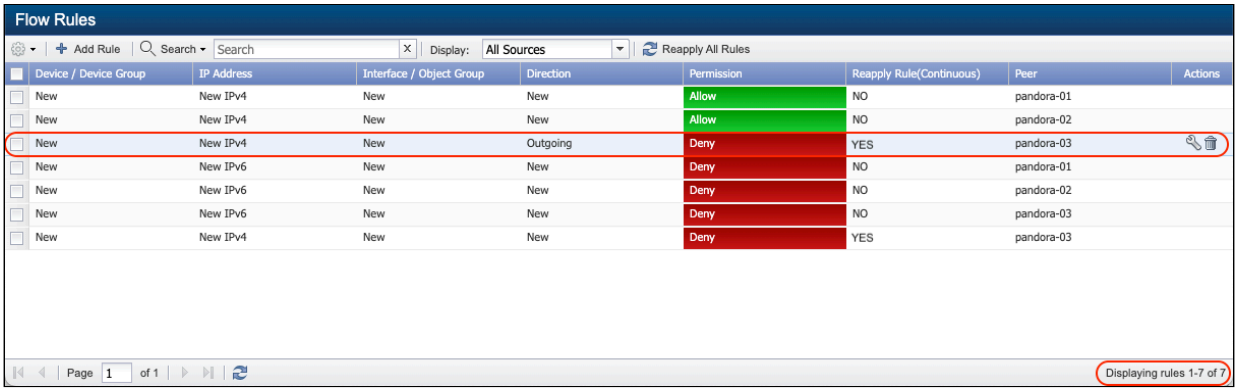
Example: Edit Flow Rule



At present, there are a total of 6 flow rules.

Click  on the row selected above.

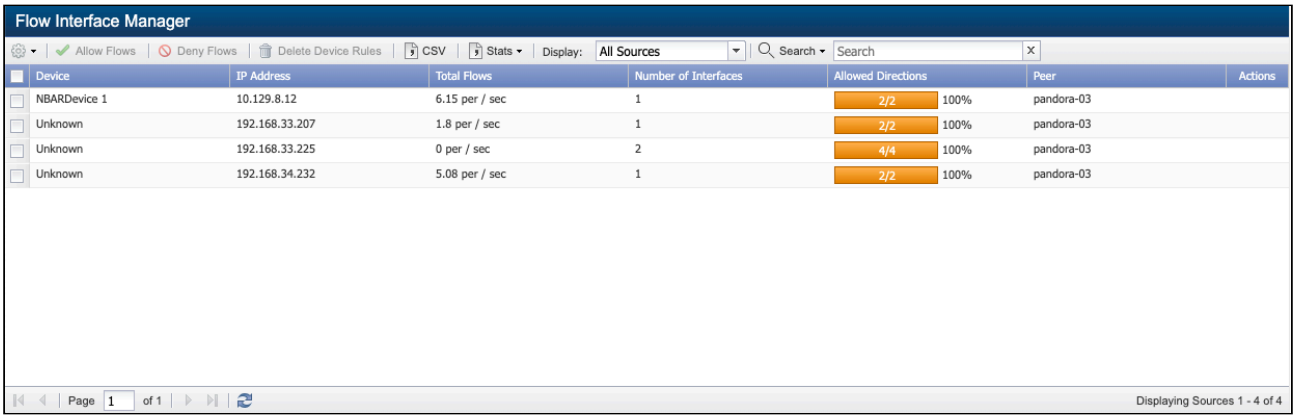
Change field **Direction** to **Outgoing**, **Permission** to **Deny**, and select **Reapply Rule(Continuous)** check box > click **Save** and refresh the page. You will notice that a new flow rule has been created and there are a total of 7 flow rules now.


Flow Rules
Add Rule
Search
Search
Display:
All Sources
Reapply All Rules
Device / Device Group
IP Address
Interface / Object Group
Direction
Permission
Reapply Rule(Continuous)
Peer
Actions
New
New IPv4
New
New
Allow
NO
pandora-01
New
New IPv4
New
New
Allow
NO
pandora-02
New
New IPv4
New
Outgoing
Deny
YES
pandora-03
New
New IPv6
New
New
Deny
NO
pandora-01
New
New IPv6
New
New
Deny
NO
pandora-02
New
New IPv6
New
New
Deny
NO
pandora-03
New
New IPv4
New
New
Deny
YES
pandora-03
Page
1
of 1
Displaying rules 1-7 of 7

43 Flow Interface Manager

The Flow Interface Manager enables you to limit the flow data that SevOne NMS processes from specific devices and from specific interfaces. When you enable devices to send flow data to SevOne NMS, SevOne NMS allows and processes all flow data by default. Devices have the potential to send large amounts of flow traffic. The rules you define here override the global flow rules you define on the Flow Rules page.

To access the Flow Interface Manager from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **Flow Interface Manager**.

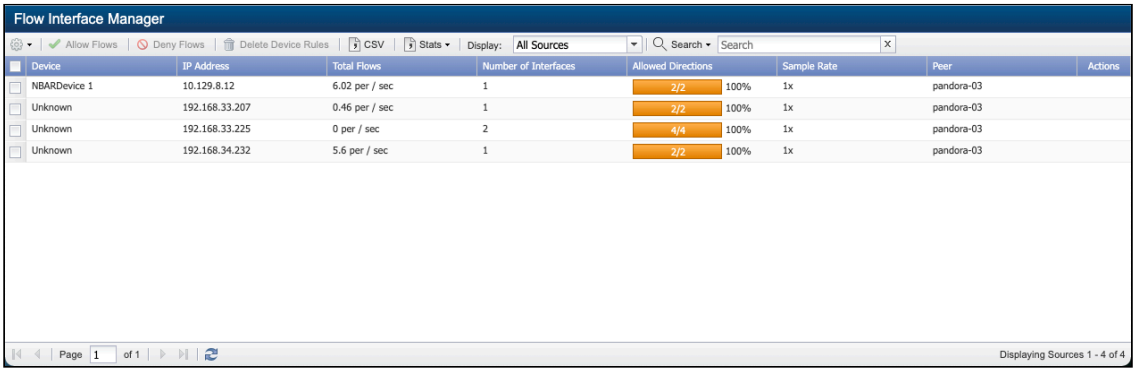


The list displays the following information for all devices from which SevOne NMS can receive flow data. Click the **Display** drop-down to display rules for **All Sources**, **Allowed Sources**, or **Denied Sources**.

- **Device** – Displays the name of the device when SNMP resolvable. Displays *Unknown* if you do not enable the SNMP plugin for the device.
- **IP Address** – Displays the IP address of the device.
- **Total Flows** - Displays the number of flows processed per second across all interfaces on the device over the past minute. Malformed flows and flows denied by a rule are not processed. The flow rate on the Flow Interface Manager is calculated after duplication.

  > The Flow Interface Manager displays the rate of flows over the past minute for each interface and direction after SevOne NMS duplicates flows that lack directional information. Since NetFlow v5 only exports information about the incoming interface, SevOne NMS duplicates the flow statistics for v5 NetFlow to factor for outgoing flows on devices that use v5 NetFlow. Therefore, if your network only uses v5 NetFlow, you can expect the flow rate to be double the actual rate of flows that arrive. The flow rate on the Flow Interface Manager is therefore different from the flow rates that display in FlowFalcon reports and on the Cluster Manager, Peer Overview tab which use different calculations.

- **Number of Interfaces** – Displays the number of interfaces on the device from which flow data is received.
- **Allowed Direction** - Displays the number of interfaces from which flow data is processed and the number of directions of flow data received. Each interface can have incoming flow and outgoing flow and you can define rules to deny flow by direction.
- **Sample Rate** - Displays the flow data sample rate when the interface sends sampled flow data. This column is only available when you select the **Display Flow Sample Rates** check box on the Cluster Manager > **Cluster Settings** tab > **FlowFalcon** subtab.
  - *n/a* – Flow data has yet to be received from the interfaces.
  - 1x - Sample rate is 1-to-1 (data is not sampled).
  - <n>x – The sample rate (e.g., if 1 packet out of 100 packets is received, this column displays 100x).

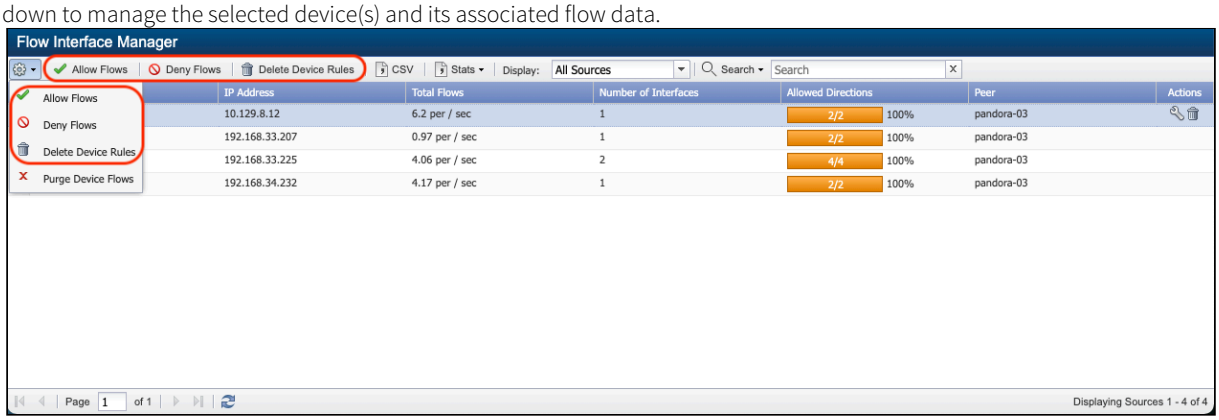


> Some flow devices only record data for a selection of messages that the device encounters based on a sample flow rate. The device notifies monitoring systems about only a fraction of its total traffic. The sample rate enables SevOne NMS to scale the data to compensate for the lack of notification of sampled data. The Sample Rate column is only available when you select the **Display Flow Sample Rates** check box on the Cluster Manager > **Cluster Settings** tab > **FlowFalcon** subtab.

- **Peer** - Displays the name of the peer that receives the flow data.

43.1 Manage Device Level Flows

- Select one or more devices and the following highlighted controls are available from the navigation bar and  drop-down to manage the selected device(s) and its associated flow data.



  - **Allow Flows** - to process the flow data across all interfaces on the selected devices.
  - **Deny Flows** - to not process the flow data for the selected devices.
  - **Delete Device Rules** - to delete the selected flow device(s) and its associated flow data.

> To be able to delete a device from Flow Interface Manager, the incoming flows must be stopped from the device being deleted. Otherwise, it will be immediately be recreated and not deleted.

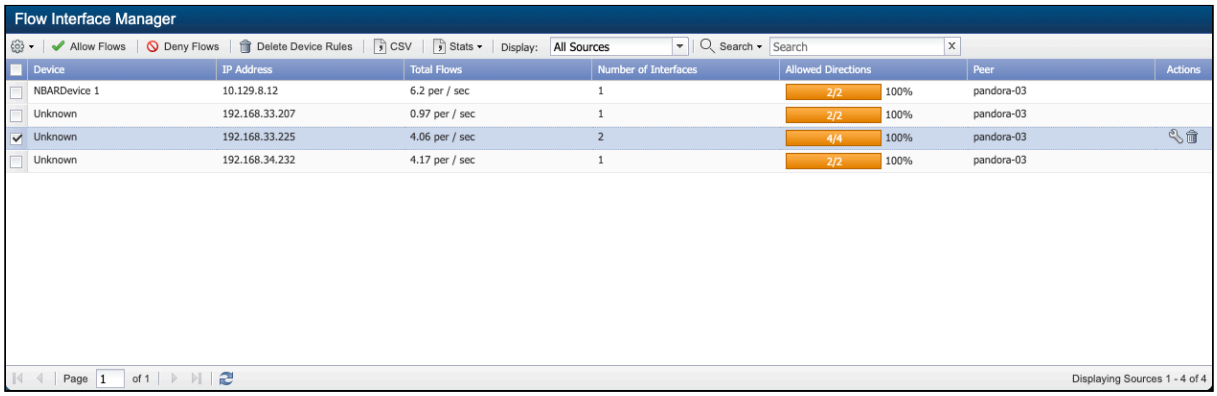
- **CSV** - to create a .csv report on all devices with flow. This includes such details as peer name, flows per second, maximum sample rate, interface, etc.
- **Stats** - click the drop-down and select **Selected Devices** to view statistics for selected devices or **All devices** to view statistics for all devices. This creates a .csv file with information such as number of accepted flows, number of dropped flows by duration, total number of dropped flows, etc.
- **Display** - click the drop-down to display rules for **All Sources**, **Allowed Sources**, or **Denied Sources**.
- Click  and select **Purge Device Flows** to delete the flow data processed for the devices.

> Alternatively, to manage a device and its associated flow data, right-click on the row of the device you want to manage. The following options are available.

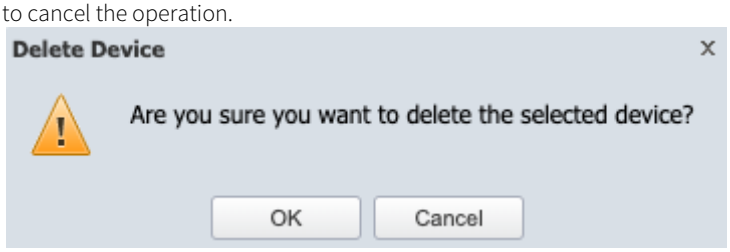
- If flow data is not being processed for the selected device, select **Allow Selected Flows** to process its flow data. If flow data is being processed and you want to stop processing it for the selected device, choose **Deny Selected Flows**.
- **Delete Selected Device Flows** - to delete the selected device and its associated flow data. If **Administration** > **Flow Configuration** > **Flow Rules** > field **Permissions** is set to **Allow**, the flow interface for the device will be recreated if it receives the flow data. If you do not want to receive the flows, set the **Permissions** field to **Deny**.
- **Purge Selected Device Data** - to delete the flow data processed for the selected device.

43.2 Delete Device Level Flow

- Select a device from the list or hover in the **Actions** column on a row for the device and its associated flow data you wish to delete. Click on  to delete the flow device and its associated flow data.



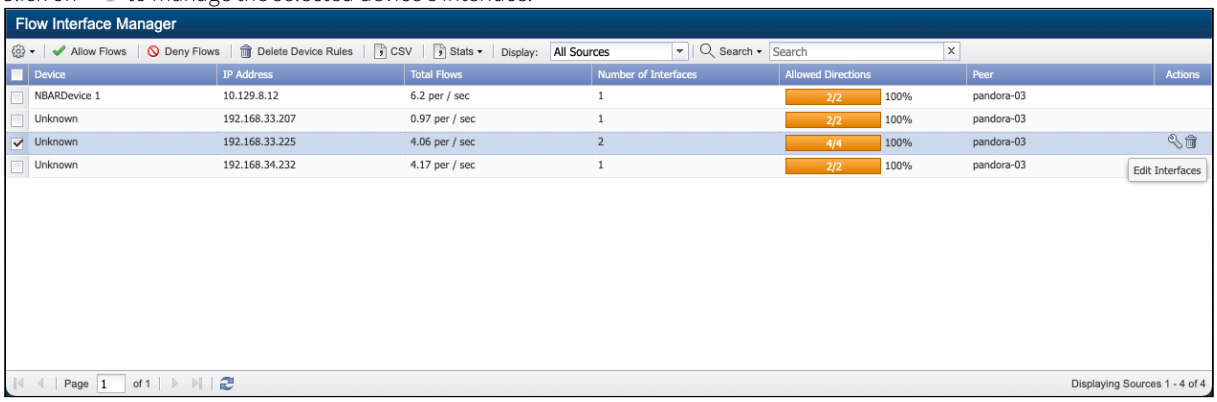
- Click on **OK** in the warning message pop-up if you are sure you want to delete the selected flow device. Click on **Cancel** or **x** to cancel the operation.



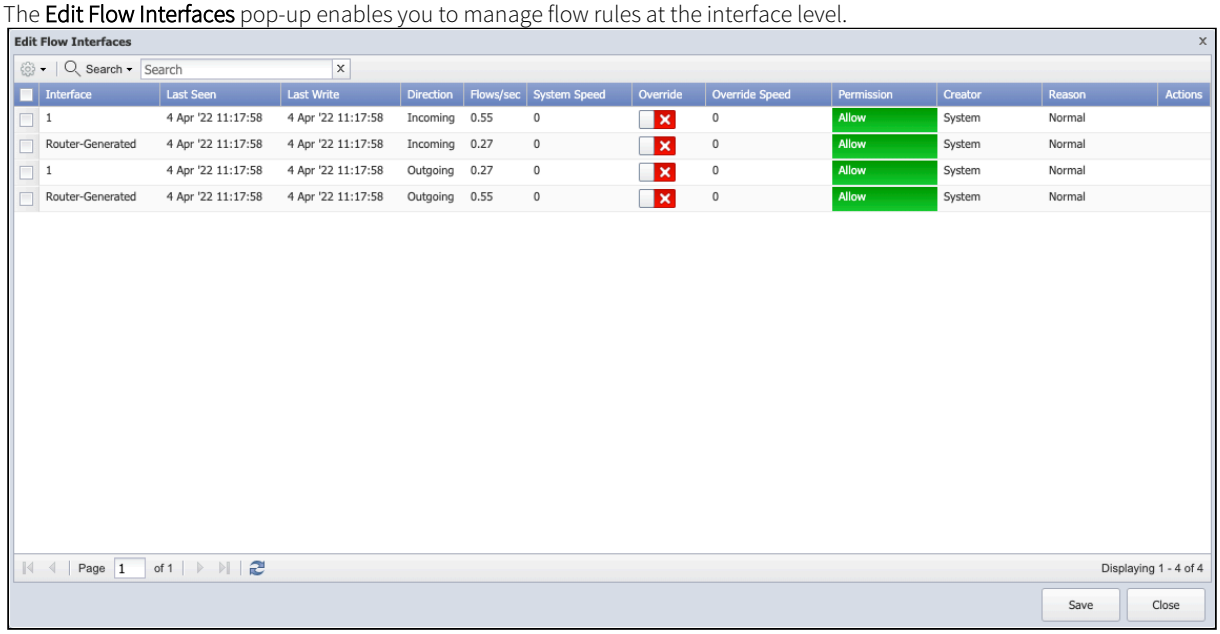
This allows deletion of only one selected flow device at a time.

43.3 Manage Interface Level Flow

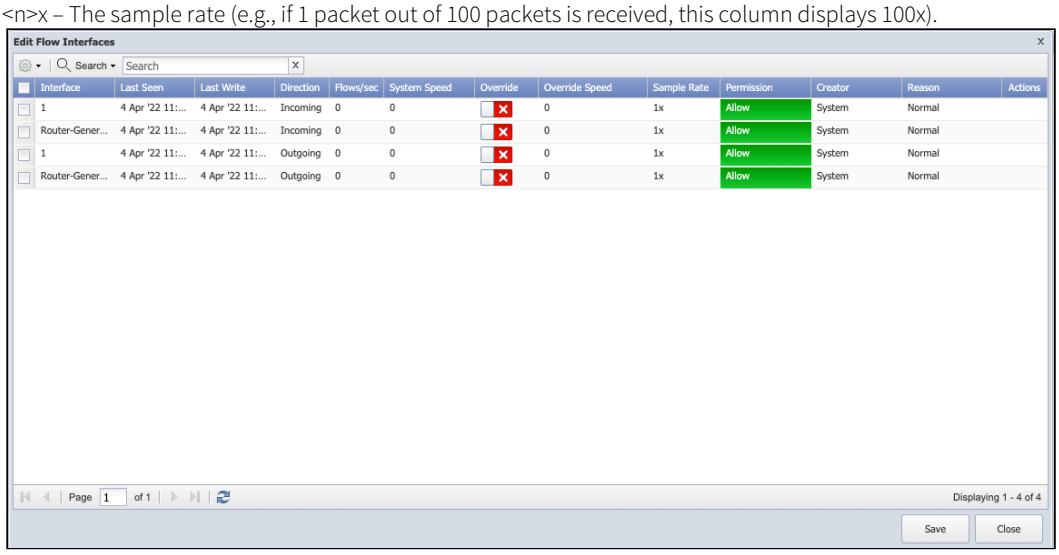
- Select a device from the list or hover in the **Actions** column on a row for the device whose interface you want to manage. Click on  to manage the selected device's interface.



- The **Edit Flow Interfaces** pop-up enables you to manage flow rules at the interface level.



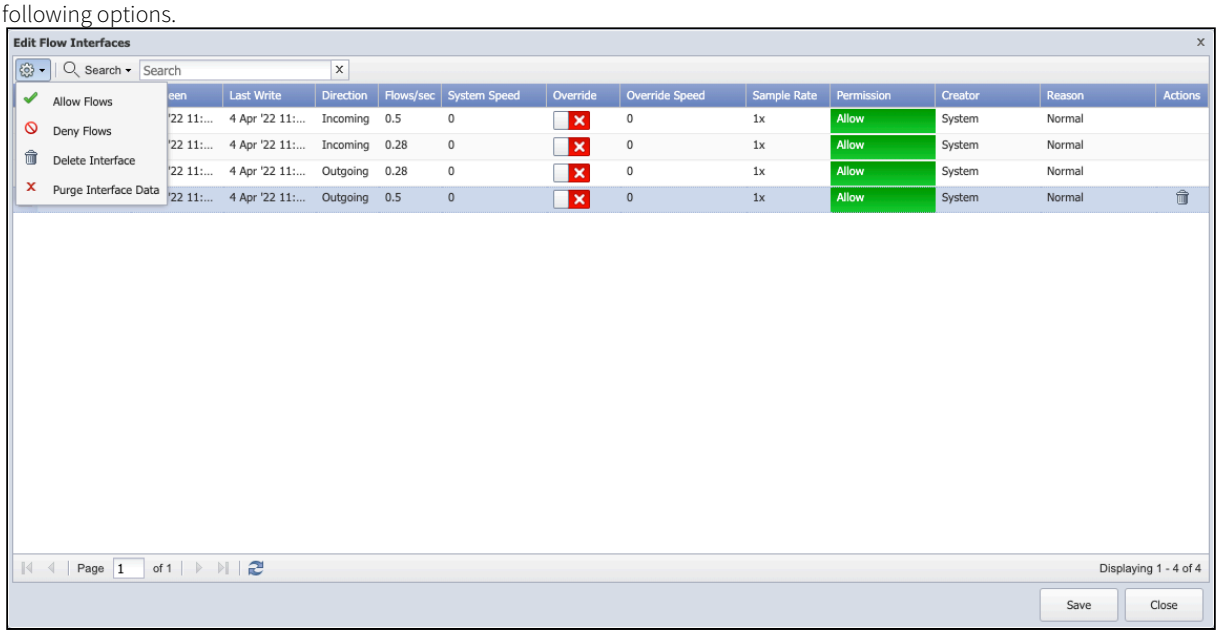
- The list displays the following information for each individual interface on the selected device.
    - **Interface** - Displays the interface number the device sends to SevOne NMS.
    - **Last Seen** - Displays the last time flow data passed through the interface.
    - **Last Write** - Displays the last time flow data from this interface was written to the database. This is either the last time flow data was received for the interface or the last time SevOne NMS wrote flow data to the database based on the Write Interval you define on the Cluster Manager > **Cluster Settings** tab > **FlowFalcon** subtab.
    - **Direction** - Displays *Incoming* for incoming flow data or displays *Outgoing* for outgoing flow data.
    - **Flows/Sec** - Displays the number of flows processed per second across the interface over the past minute.
    - **System Speed** - Displays the system discovered speed associated with the automatically mapped metric object / indicator.
    - **Override** - To allow user to turn on override to change the override speed.
    - **Override Speed** - To allow user to enter the override speed.
    - **Sample Rate** - Displays the flow data sample rate when the interface sends sampled flow data.
        - *n/a* – Flow data has yet to be received from the interfaces.
        - 1x - Sample rate is 1-to-1 (data is not sampled).
        - <n>x – The sample rate (e.g., if 1 packet out of 100 packets is received, this column displays 100x).



> The sample rate enables SevOne NMS to scale the data to compensate for the lack of notification of sampled data. The Sample Rate column is only available when you select the **Display Flow Sample Rates** check box on the Cluster Manager > **Cluster Settings** tab > **FlowFalcon** subtab.

- **Permission** - Displays *Allow* when data is processed across the interface. Displays *Deny* when data is not processed across the interface.
- **Creator** - Displays *System* when SevOne NMS creates the interface or a FlowFalcon Interface rule updates the interface. Displays *User* when a user creates or updates the interface.
- **Reason** - Displays *Normal* when data can be processed across the interface. Displays *Exceeds Capacity* when the object count exceeds the peer license capacity and flows cannot be processed for the interface. For licensing purposes, each interface is equal to 300 objects.

- To manage the interface(s), select one or more interface from the list. Click  drop-down and select one of the following options.



  - **Allow Flows** - to process the flow data across the selected interface(s).
  - **Deny Flows** - to not process the flow data across the selected interface(s).
  - **Delete Interface** - to delete the selected flow device interface(s) and its associated flow data.

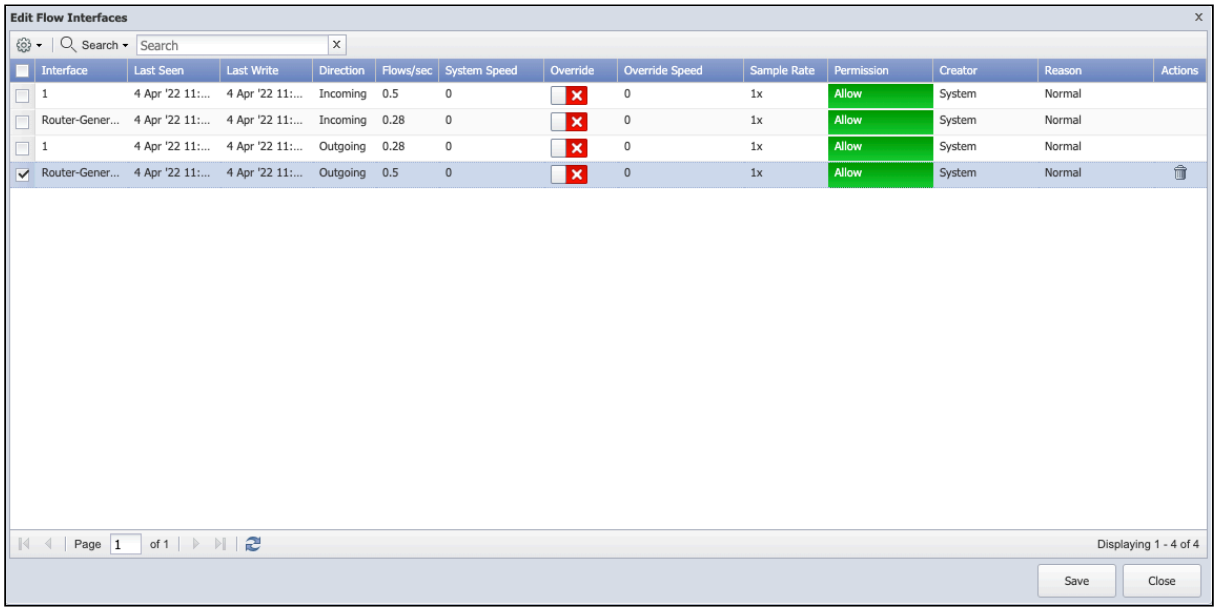
    > ⚠ To be able to delete an interface from a device, the flows received, related to this device, must not contain information for the interface to be deleted. Otherwise, it will be automatically recreated and not deleted.

  - **Purge Interface Data** - to delete the flow data for the interfaces.

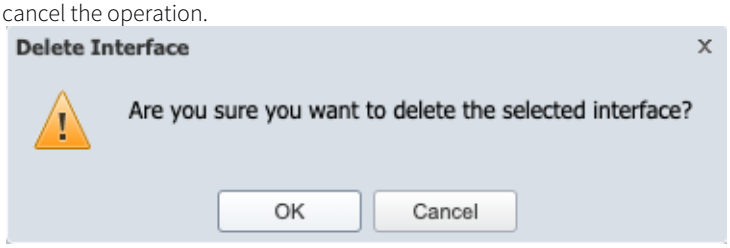
> ⚠ Alternatively, to manage an interface, right-click on the row to manage the selected device's interface. The following options are available.
>
> - If flow data is not being processed for the selected interface, select **Allow Selected Flows** to process its flow data. If flow data is being processed and you want to stop processing it for the selected interface, choose **Deny Selected Flows**.
> - **Delete Selected Flows** - to delete the selected interface and its associated flow data. If **Administration** > **Flow Configuration** > **Flow Rules** > field **Permissions** is set to **Allow**, the flow interface will be recreated if it receives the flow data. If you do not want to receive the flows, set the **Permissions** field to **Deny**.
> - **Purge Selected Data** - to delete the flow data processed for the selected interface.

43.4 Delete Interface Level Flow

- Select an interface from the list or hover in the **Actions** column on a row for the interface you want to delete. Click on  to delete the selected interface and its associated flow data.



- Click on **OK** in the warning message pop-up if you are sure you want to delete the selected interface. Click on **Cancel** or **x** to cancel the operation.



This allows deletion of only one selected interface at a time.

44 MPLS Flow Mapping

The MPLS Flow Mapping page enables you to upload your network's MPLS flow map files. MPLS flow map files map MPLS attributes to flow data so that MPLS data can appear in FlowFalcon reports. To allow reporting, network-specific MPLS network attributes such as, VPN 2nd Top Layer ID, PE Egress Address, Customer VRF, Source IP Address, and PE Ingress Address are required. Use any application to create two .csv files that map MPLS attributes to flow data. Then use this page to upload the map files into SevOne NMS. The .csv files must be encoded in UTF-8.

About this feature...

MPLS attribute mapping feature is designed to help you report on flow data exported from the MPLS network that is used by multiple tenants and, the flow data can be associated with different tenants. It enriches flow data that includes the standard MPLS fields with attributes like Customer VRF Name and PE Ingress IP to enhance reporting and help users derive better insights.

This feature is *optional* for reporting on MPLS flow data. The mapping tables are only required for the MPLS attribute mapping feature.

In scenarios where no MPLS attribute mapping is needed, no special configuration is required to support flexible NetFlow with MPLS fields. A custom flow view is needed to report on the MPLS flow data.

From **Cluster Manager** > **Cluster Settings** tab > **FlowFalcon** subtab > field **Enable MPLS Attribute Mapping** allows you to map v9 NetFlow template data from core "P" routers and provides flow data for *Customer Client IP*, *Customer Client Subnet*, *Customer VRF*, *Customer Application IP*, *Customer Application Subnet*, *PE Ingress IP*, and *PE Egress IP*. It maps tenant names to conversations exported in flow records from core "P" routers using information from the .csv files.

The two mapping .csv files can be prepared using *VRF Name* and *PE Ingress IP* mappings based on information from your network. This mapping information can be collected via SNMP from the PE devices and other methods.

In addition to the standard 5-tuple attributes, the following attributes present in the Netflow v9 template data are used to perform a lookup in the mapping tables contained in the .csv files.

- "MPLS-Label2" or mplsLabelStackSection2 - Customer VRF Label ID
- "TopLabelAddr" or mplsTopLabelIPv4Address - PE Egress IP Address
- "SrcAddr" - Customer Source IP Address
- "DstAddr" - Customer Destination IP Address

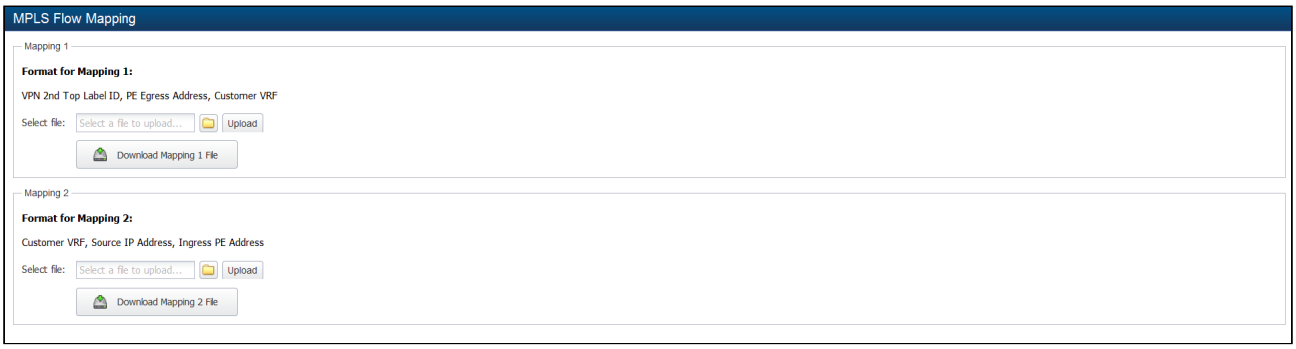
When field **Enable MPLS Attribute Mapping** is enabled, all received flows are enriched with the following fields.

Field	Description
Customer Client IP	Customer-specific IP address of connection origin.
Customer Client Subnet	Customer-specific IP subnet for connection origin.
Customer VRF	Name of Customer VRF.
Customer Application IP	Customer-specific IP address for connection target.
Customer Application Subnet	Customer-specific IP subnet for connection target.
PE Ingress IP	IP address of Ingress PE router.
PE Egress IP	IP address of Egress PE router.

On SevOne **DNC** appliances, *two* additional mapping tables are maintained to add the enriched fields as .csv files.

- (VPN 2nd Top Label ID, PE Egress Address) mapped to Customer VRF.
- (Customer VRF, Source IP Address) mapped to PE Ingress Address.

To access the MPLS Flow Mapping page from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **MPLS Flow Mapping**.



In SevOne NMS, there are three requirements to map MPLS attributes to flow data for FlowFalcon reports.

- On the Cluster Manager > **Cluster Settings** tab, **FlowFalcon** subtab, select the **Enable MPLS Attribute Mapping** check box and enter the **MPLS Attribute Mapping Refresh Interval**.
- On the **MPLS Flow Mapping** page, upload two MPLS mapping files.
- On the FlowFalcon View Editor, create FlowFalcon views that include at least one of the following fields:
  - 45050: Customer Client IP
  - 45051: Customer Client Subnet
  - 45052: Customer VRF Name
  - 45053: Customer Application IP
  - 45054: Customer Application Subnet
  - 45055: PE Ingress IP
  - 45056: PE Egress IP

44.1 Upload Map Files to DNC

The map files must be uploaded to a single SevOne Dedicated NetFlow Collector (DNC) that uses the file. Because each DNC may be responsible for different mappings, it is necessary to upload a mapping file to each DNC that performs mapping. In order to direct the mapping file to a DNC, first navigate to the IP of the DNC, then upload the file.

44.2 Upload Map Files in SevOne NMS

Perform the following steps to upload the two files that map MPLS attributes. The first map file maps VPN 2nd Top Label ID, PE Egress Address, Customer VRF. The second map file maps Customer VRF, Source IP Address, PE Ingress Address.

1. In the **Mapping 1** section, click  to display the File Upload pop-up.
2. Navigate the file structure to locate and select the file that maps the VPN 2nd Top Label ID, PE Egress Address, and the Customer VRF.
3. Click **Open** on the pop-up to save the file locally.
4. Click **Upload** to move the file to the correct location and to complete the upload of the first map file.
5. Click **Download Mapping 1 File** to display the content of the first map file in a .csv format.

   When you click **Download Mapping 1 File** button, it downloads a default **VRFMapping.csv** file with the format this feature expects the data to be in.

   The .csv file uses Field ID 47 (*MPLS Top Protocol IP - mplsTopLabelIPv4Address*) and Field ID 71 (*MPLS Stack Entry 2 - mplsLabelStackSection2*) to lookup the mapped VRF Name. Field IDs **47** and **71** make use of the mapping CSV table to report on *PE Egress IP*, *PE Ingress IP*, and *VRF Name* in the flow reports.

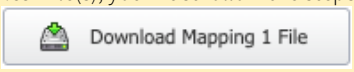
6. In the Mapping 2 section, click  to display the File Upload pop-up.
7. Navigate the file structure to locate and select the file that maps the Customer VRF, Source IP Address, and PE Ingress Address.

8. Click **Open** on the pop-up to save the file locally.
9. Click **Upload** to move the file to the correct location and to complete the upload of the second map file.
10. Click **Download Mapping 2 File** to display the content of the second map file in a .csv format.

> When you click **Download Mapping 2 File** button, it downloads a default **PEMapping.csv** file with the format this feature expects the data to be in.

> The .csv file uses the VRF Name, and Field ID 8 (*Source IP*) or Field ID 27 (*Source IPv6*) to lookup the mapped PE Ingress IP and report on *PE Egress IP*, *PE Ingress IP*, and *VRF Name* in the reports using flow views that include these fields

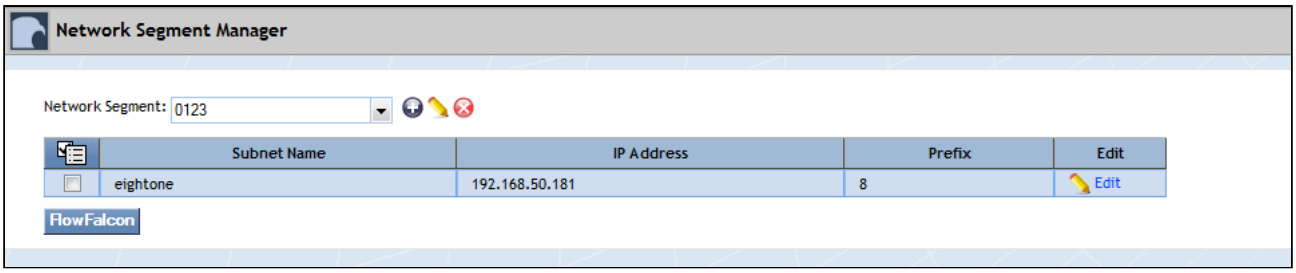
> The two mapping .csv files can be prepared by using VRF Name and PE Ingress IP mappings based on the information available from your network.

> **Modify .csv file(s)**
>
> To modify the .csv file(s), you must follow the steps below.
>
> a. Click 
>
> and/or 
>
> button to download the VRF / PE .csv mapping file(s) respectively.
> b. Edit the .csv file(s) you have downloaded.
> c. Click  to navigate to the .csv file you want to upload then, click 
>
> button.
> d. Based on value set in field **MPLS Attribute Mapping Refresh Interval** from Cluster Manager > **Cluster Settings** tab, **FlowFalcon** subtab, the updates made in .csv file(s) will be applied by the **netflowd** collector. Field **MPLS Attribute Mapping Refresh Interval** is only available when **Enable MPLS Attribute Mapping** check box is selected.

45 Network Segment Manager

The Network Segment Manager enables you to define network segments to group flow data. You group subnets into the network segments to enable you to identify the traffic that comes from the group of networks.

To access the Network Segment Manager from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **Network Segment Manager**. The FlowFalcon Reports page also provides access to the Network Segment Manager.



45.1 Manage Network Segments

You can create network segments that contain multiple subnets with the same name to group flow data. The network segments you define here appear on the Report Attachment Wizard and on the FlowFalcon Reports page in the Network Segment drop-down list. Please refer to *SevOne NMS User Guide* for details on these pages.

1. Click the **Network Segment** drop-down and select a network segment. The subnets for the segment you select appear in the list below.
2. Click the following icons to define segments.
   - - Click to display the Network Segment pop-up where you enter the name of a new network segment.
   - - Click to display the Network Segment pop-up to where you change the name of a network segment.
   - - Click to delete a network segment.
3. View the list of the subnets in the segment.

45.2 Manage Subnets

The list displays the subnets in the network segment you select.

1. Click and select **Add New Subnet** or click **Edit** next to display the Subnet pop-up.
2. In the **Subnet Name** field, enter the subnet name. When you create more than one subnet with the same name, the data from those subnets is combined in reports.

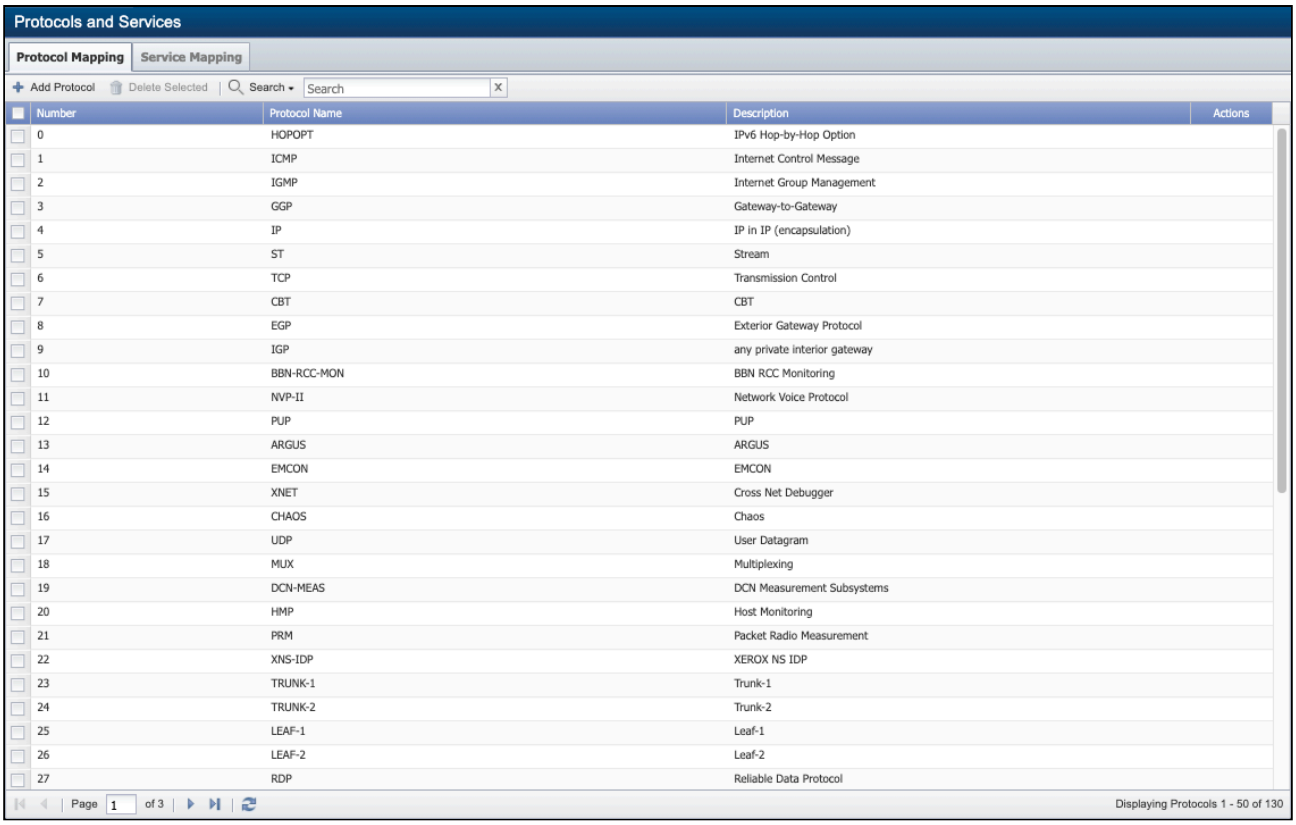
   Example: You have subnets, 192.168.30.0/24 and 192.168.20.0/24. You name both subnets *Web Servers*. FlowFalcon reports combine the traffic from both Web Server subnets and display one result.
3. In the **IP Address** field, enter the subnet IP address.
4. In the **Prefix** field, enter the subnet prefix (also referred to as the CIDR address, network mask, or number of borrowed bits e.g., /24).
5. Click the **Network Segment** drop-down and select the network segment to which to associate the subnet.
6. Click **Save**.

   When you add a new subnet (or edit an existing subnet), the canonical IP address for the subnet will appear in the table based on the IP address and prefix that you provide. For example, if you provide the IP address **192.168.10.1** and the prefix **8**, the IP address that will appear for the network segment would be **192.0.0.0**.

46 Flow Protocols and Services

The Protocols and Services page enables you to edit and define new protocols and services from which SevOne NMS can collect flow data.

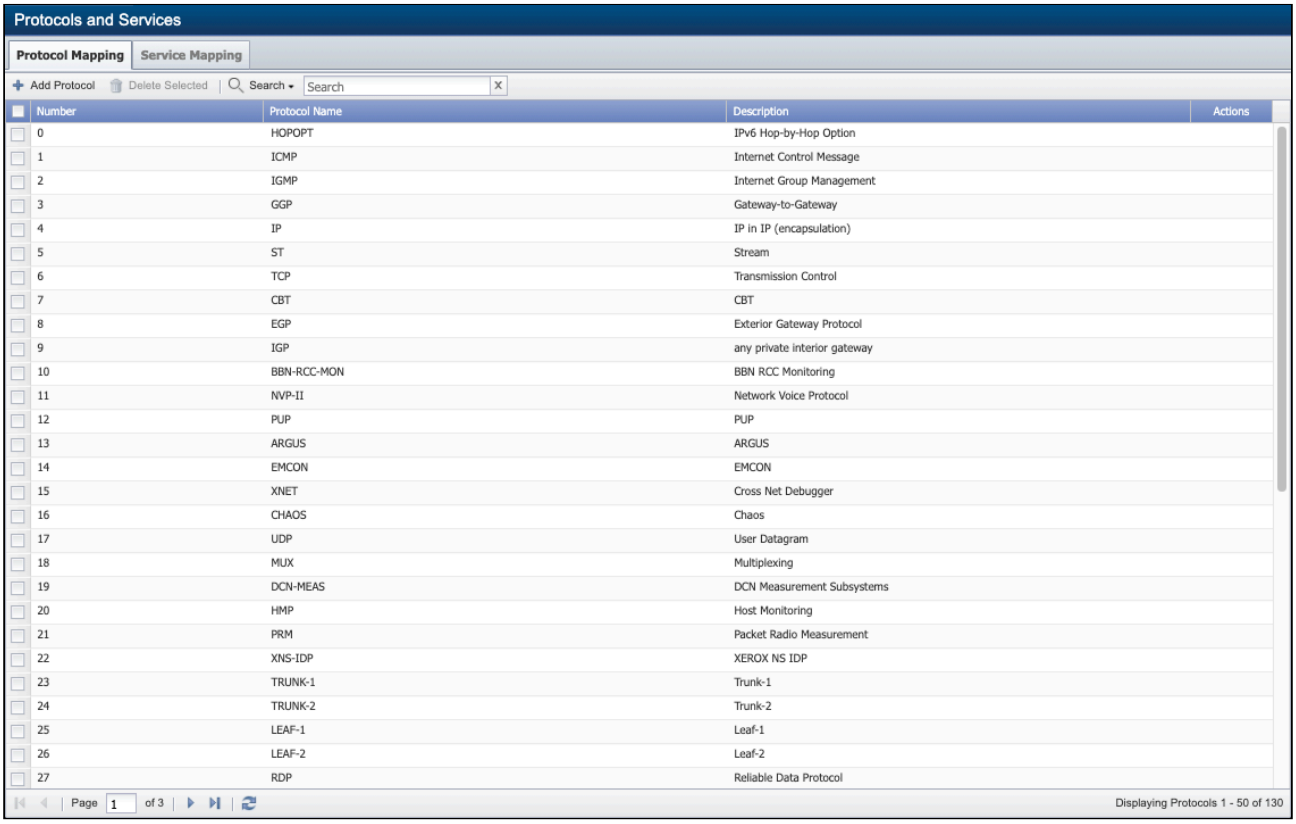
To access the Protocols and Services page from the navigation bar, click the **Administration** menu, select **Flow Configuration**, and then select **Protocols and Services**.



The Protocol Mapping tab lists the protocols for which you can create a flow report and the Service Mapping tab lists the services for which you can create a flow report.

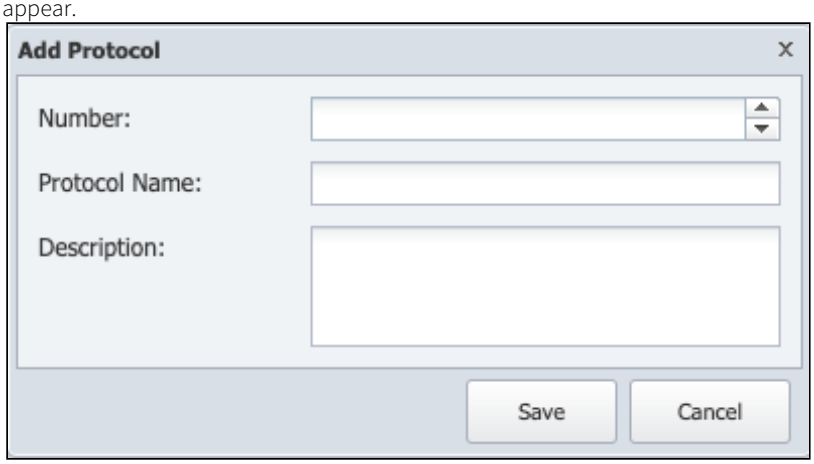
46.1 Manage Protocol Mapping

The Protocol Mapping tab displays the flow protocols SevOne NMS discovers.



46.1.1 Add Protocol

1. Click **Add Protocol** to add a protocol or click  to edit an existing protocol. **Add Protocol** / **Edit Protocol** pop-up will appear.



2. In the **Number** field enter the protocol number.
3. In the **Protocol Name** field, enter the protocol name.
4. In the **Description** field, enter the protocol description.
5. Click **Save**.

46.1.2 Delete Selected

Select the check box for each protocol to be deleted. Click **Delete Selected** to delete.

> **Delete Selected** button is only available when at least one protocol is selected to be deleted.

46.1.3 Search

From **Search** drop-down, enable **Select All Columns** to allow you to search both **Protocol Name** and **Description** columns for the text entered in the search box. You have an option to search for text in either **Protocol Name** or **Description** column based on which option is selected from the Search drop-down.

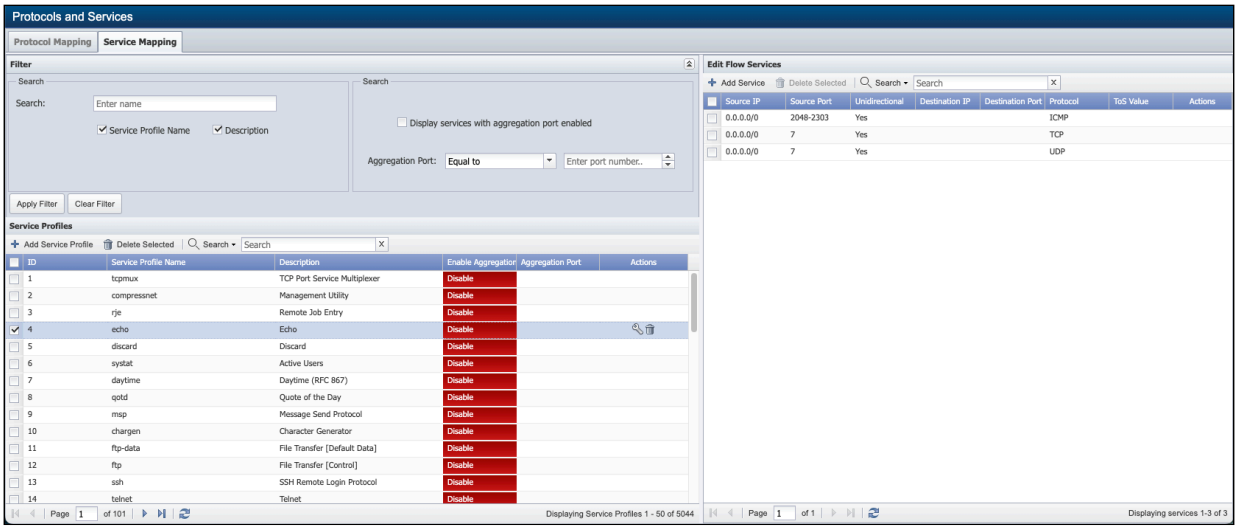
- Search is case-insensitive.
- Search cannot be performed on the **Number** column.
- At least one character is required to do the search on.

46.2 Manage Service Mapping

IMPORTANT

Modifications to flow services can take up to 5 minutes for report time resolution to take effect.

The Service Mapping tab displays the flow services SevOne NMS discovers.



Each service has a number of matching rules associated with it and these matching rules are used to match the flow as it arrives back to a service profile. In the screenshot above, you see that Service Profile ID = 4 has three flow services, as shown in the right pane.
The Service Profile ID is stored in the flow itself. FlowFalcon View Editor manages the FlowFalcon views used to create reports, aggregated and raw, using the Service Profile id.

46.2.1 Filter

Filters enable you to limit the services that appear in the list. Filters are optional.

46.2.1.1 Search

The Search section allows the search capability based on the following.

- In the **Search** field, enter text you want to search on. Select **Service Profile Name** and/or **Description** check boxes to perform the search in service profile name and/or description column(s) for the text entered after the filter is applied.
- Select the **Display services with aggregation port enabled** check box to filter on services that have **Enable Aggregation Port** set to **Enable**.

- Click the **Aggregation Port** drop-down and choose from options **Equal to**, **Less than**, or **Greater than**. Enter the port number in the text field to perform the search in the aggregation port column based on the option chosen.

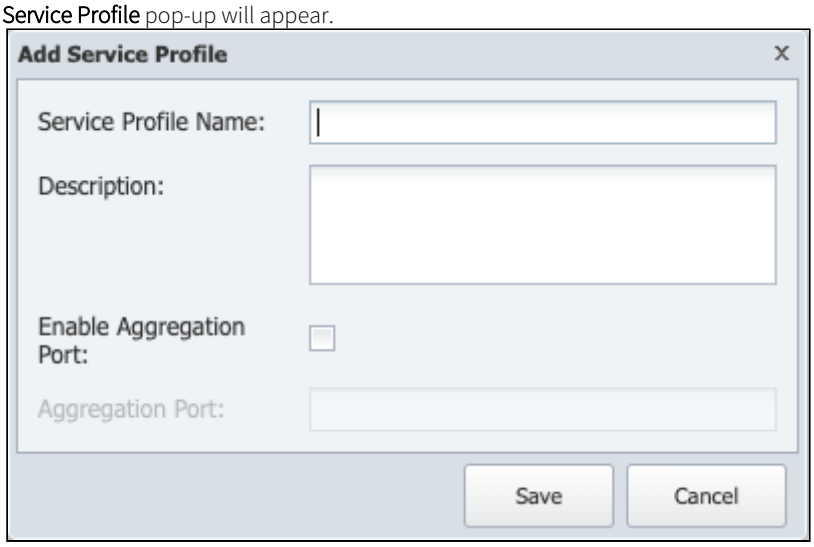
46.2.1.2 Buttons

- Click **Apply Filter** button to apply the filter settings.
- Click **Clear Filter** button to remove all filters and to display all flow services in the list.
- Click on  to collapse or  to uncollapse the Filter section.

46.2.2 Service Profiles

46.2.2.1 Add Service Profile

1. Click **Add Service Profile** to add a service profile or click  to edit an existing service profile. **Add Service Profile** / **Edit Service Profile** pop-up will appear.



2. In the **Service Profile Name** field, enter the service profile name to appear in reports.
3. In the **Description** field, enter the service profile description.
4. Click the check box to enable **Enable Aggregation Port** and enter the port number in **Aggregation Port** field.

> **Enable Aggregation Port** check box when enabled,
> - applies the **Aggregation Port** to aggregated and raw flow data.
> - the application port is rewritten for both aggregated and raw data.
> - aggregated and raw data retain the integrity of the original non-application ports; i.e., source, destination, and client ports.

5. Click **Save**.

46.2.2.2 Delete Selected

Select the check box for each service to be deleted. Click **Delete Selected** to delete.

> **Delete Selected** button is only available when at least one service profile is selected to be deleted.

46.2.2.3 Search

From **Search** drop-down, enable **Select All Columns** to allow you to search all columns such as, **ID**, **Service Profile Name**, **Description**, and **Aggregation Port** for the text entered in the search box. You have an option to search for text in **ID** or **Service Profile Name** or **Description** or **Aggregation** columns based on the option selected from the Search drop-down.

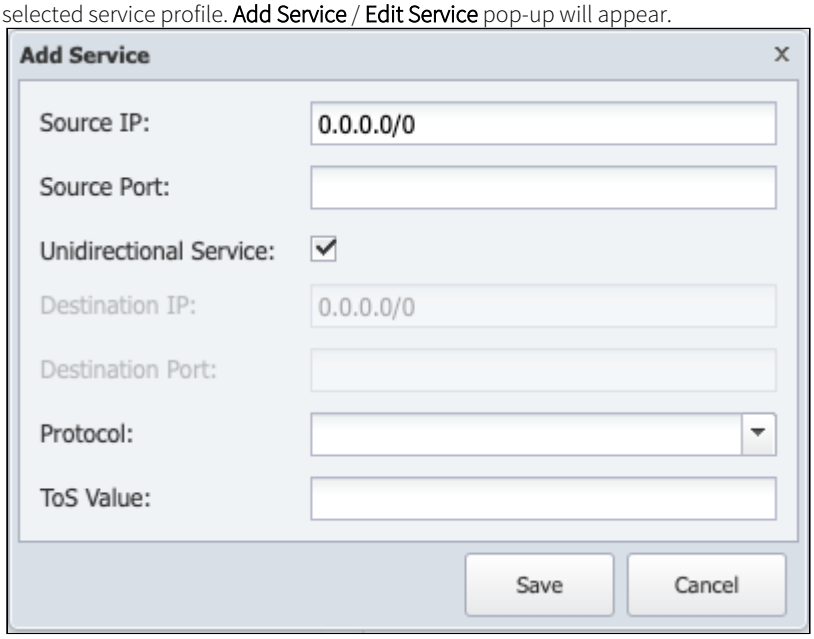
- Search is case-insensitive.
- Search cannot be performed on the **Enable Aggregation Port** column.
- At least one character is required to do the search on.

46.2.3 Edit Flow Services

From **Service Profiles**, select a service profile name to view its available flow services.

46.2.3.1 Add Service

1. Click **Add Service** to add a service to the service profile selected or click to edit an existing service available to the selected service profile. **Add Service** / **Edit Service** pop-up will appear.



2. In the **Source IP** field, enter the source IP address.
3. In the **Source Port** field, enter the source port number.
4. Select the **Unidirectional Service** check box to allow service to be unidirectional only. For bidirectional service, disable the check box and configure the following fields.
    a. In the **Destination IP** field, enter the destination IP address.
    b. In the **Destination Port** field, enter the destination port number.
5. In the **Protocol** field, select a protocol from the drop-down list.
6. In the **ToS Value** field, enter the value for the type of service.
7. Click **Save**.

46.2.3.2 Delete Selected

Select the check box for each flow service to be deleted. Click **Delete Selected** to delete.

Delete Selected button is only available when at least one flow service is selected to be deleted.

46.2.3.3 Search

From **Search** drop-down, enable **Select All Columns** to allow you to search all columns such as, **Source IP**, **Source Port**, **Unidirectional**, **Destination IP**, **Destination Port**, **Protocol**, and **ToS Value** for the text entered in the search box. You have an option to search for text in **Source IP** **Source Port** or **Unidirectional** or **Destination IP** **Destination Port** or **Protocol** or **ToS Value** columns based on the option selected from the Search drop-down.

- Search is case-insensitive.
- At least one character is required to do the search on.

47 Enable Flow Technologies

If your SevOne appliance has more than one Network Interface Card (NIC) run the following command one time.

```
echo 0 > /proc/sys/net/ipv4/conf/all/rp_filter
```

47.1 Send Flow Data To SevOne NMS

This topic describes how to enable flow devices to send flow data to SevOne NMS. This workflow is outside of the SevOne NMS application and may not present all of the steps your network requires to enable devices to send flow data. If the following instructions are not applicable for your network please reference the device manufacturer's documentation.

This is a brief list of devices and the corresponding commands to set up flow. If your device is not in this list, it does not mean SevOne NMS does not support your device. Contact the device vendor for instructions to enable flow. Only people with Cisco or similar device configuration experience should perform flow setup.

47.1.1 Flow Source Flow Timeout Configuration

The typical manufacturer setting is for a router to send flow data every 30 minutes. Sometimes referred to as the flow cache timeout, this setting defines the frequency that a router sends the flow table to the collector (SevOne NMS). This implicitly is the limit to which the router allows a flow to grow before breaking the flow into a new flow.

SevOne recommends that you configure routers to send flow data every one minute in order to have the router report to SevOne NMS in a timely manner that enables the even distribution of information transfer. Should you choose to set the flow source flow timeout configuration to something other than one minutes, the router reports less frequently and sends SevOne NMS larger flow tables which results in less granular report data. To compensate for this, SevOne NMS FlowFalcon reports provide a Granularity setting that enables you to view the report at the granularity that matches your router flow timeout configuration. A flow cache timeout other than one minute is not recommended.

The SevOne NMS Cluster Manager > **Cluster Settings** tab provide a Drop Long Flows option that enables you to define a time limit for what you consider to be a long flow. When you use the Drop Long Flow option, SevOne NMS hides the traffic from routers that send flows that exceed the Max Flow Duration you enter. When a router sends flows that exceed the Max Flow Duration, an administrative message appears upon log on to inform administrators that flows from a specific router have been dropped. The Drop Long Flows feature is useful when you set the router cache timeout to be shorter than the Max Flow Duration you set in SevOne NMS, because long flows would then indicate that a router is misconfigured.

The Cluster Manager > **Cluster Settings** tab provides the ability to adjust the interval at which SevOne NMS writes flow data to the database. The write interval sets the time window for which raw data is to be aggregated into the minimal aggregation. The Write Interval should be set to one minute. In the rare situation where you decide to change this setting, you should consider that every hour SevOne NMS takes flow data and creates 15 minute aggregations for the top <n> flows for each interface and view. Your Write Interval setting should therefore be divisible by 15 when you intend to use aggregated flow data.

Flow Source Flow Timeout Configuration Considerations

Applicable Use Cases	Flow Source Flow Timeout Configuration	SevOne NMS FlowFalcon Report Settings
Billing AND Bursting Monitoring (Recommended). This is the optimal SevOne NMS setting for typical NetFlow reporting	1 Minute	Leave the Display Setting Granularity set to the default "Auto".
Acceptable	2-5 Minutes	Set the Display Setting Granularity to 5 minutes

Not Recommended	5 Minute +	On the classic FlowFalcon Reports page, in the Display Settings section, click the Granularity drop-down and select Custom. Set the granularity time span to twice the router flow timeout. On the Report Attachment Wizard, on the Settings page, FlowFalcon tab, click the Granularity drop-down and select 30 minutes.

47.2 Cisco

47.2.1 Cisco IOS Router

47.2.1.1 Enable Cisco Express Forwarding

Enter the following command to enable Cisco Express Forwarding which is required for flow in most recent IOS releases.

```
router(config)# ip cef
```

47.2.1.2 Start NetFlow Export

In the configuration terminal on the router, enter the following commands to start NetFlow Data Export (NDE).

1. The address of your SevOne NMS appliance.

   ```
   router(config)# ip flow-export destination <SevOne-IP> 9996
   ```

2. The source interface is used to set the source IP address of the NetFlow exports sent by the router.

   ```
   router(config)# ip flow-export source loopback
   ```

3. Sets the export version number.

   ```
   router(config)# ip flow-export version 5 and 9
   ```

47.2.1.3 Break Up Flows into Shorter Segments

1. Breaks up long-lived flows into one minute segments.

   ```
   router(config)# ip flow-cache timeout active 1
   ```

2. Ensures the flows that have finished are exported in a timely manner.

   ```
   router(config)# ip flow-cache timeout inactive 15
   ```

47.2.1.4 Enable NetFlow on Each Physical Interface

Enter the following commands to enable NetFlow on each physical interface from which to collect a flow (not VLANs and Tunnels because they are automatically included). This is normally an Ethernet or WAN interface. You may need to set the speed of the interface in kilobits per second especially for frame relay or ATM virtual circuits.

```
router(config)# interface <interface>
router(config)# ip route-cache flow or ip flow ingress or ip route-cache cef
```

Write your configuration with the write or copy run start command.

47.2.1.5 Verify

When in enabled mode, enter the following command to view current NetFlow configuration and state.

1. Shows the current setup.

```
router# show ip flow export
```

2. Summarizes the active flows and displays how much NetFlow data the router exports.

```
router# show ip cache flow
router# show ip cache verbose flow
```

47.2.2 Cisco Switches Running CatOS (Hybrid Mode)

47.2.2.1 Non-4000 Series Catalyst Switch

47.2.2.1.1 Router Side

1. Enter the following global commands.

```
router# ip flow-export source
router# ip flow-export version 5 or 9
router# ip flow-export destination <SevOne-IP> 9996
router# ip flow-cache timeout active 1
```

2. Enter the following command on each physical interface. You must log on to each interface one at a time.

```
router# interface <interface>
router# ip route-cache flow
```

47.2.2.1.2 Switch Side

1. The address of your SevOne NMS appliance.

```
router#  set mls nde <SevOne-IP> 9996
```

2. Sets the export version.

```
router# set mls nde version 9
```

3. Breaks up long-lived flows into ~two minute segments.

```
router# set mls agingtime long 128
```

4. Ensures that flows that have finished are exported in a timely manner.

```
router# set mls agingtime 64
```

5. This sets the flow mask to full flows.

```
router# set mls flow full
```

6. CatOS 7.(2) or higher is required for this command, which enables NDE for all traffic within the specified VLANs rather than just inter-VLAN traffic.

```
router# set mls bridged-flow-statistics enable
```

7. Enables NDE.

```
router# set mls nde enable
```

47.2.2.2 Switches Running IOS (Native Mode)

Enter the following global commands (all commands are entered in the router <enable> config option).

1. Sets the export version.

```
router# ip flow-export source
router# ip flow-export version 9
router# ip flow-export destination <SevOne-IP> 9996
router# mls nde sender version 9
```

2. Breaks up longlived flows into oneminute segments.

```
router# mls aging long 64
```

3. Ensures that flows that have finished are exported in a timely manner.

```
router# mls aging normal 64
```

4. If you have a Supervisor Engine 2 or 720 running IOS version 12.1.13(E) or higher the next two commands are required to put interface and routing information into the NetFlow exports.

```
router# mls flow ip interface-full
router# mls nde interface
```

5. Enter the following command on each physical interface. You must log on to each interface one at a time.

```
router# interface <interface>
router# ip route-cache flow
```

By default, all flows are **Router Generated**. However, when **match interface input** and **match interface output** are added to the device configuration, it results in interface index information to be emitted.

47.2.3 4000 Series Catalyst Running in Hybrid or Native Mode

This series requires a Supervisor Engine IV with a NetFlow Services daughter card to support NDE.

47.2.3.1 Start NetFlow Export

In the configuration terminal on the router, enter the following command to start NetFlow export.

```
router# ip flow-export version 9
router# ip flow-export destination <SevOne-IP> 9996
```

47.2.3.2 Enable NetFlow on Each Physical Interface

Enter the following command to enable NetFlow on each physical interface.

```
router# interface <interface>
router# ip route-cache flow infer-fields
```

47.3 Juniper

Juniper supports flow exports by sampling packet headers with the routing engine and aggregating them into flows. Packet sampling is achieved by defining a firewall filter to accept and sample all traffic, applying that rule to an interface, and then configuring the sampling forwarding option.

To configure inline flow monitoring, include the `inline-iflow` statement at the `[edit forwarding-options sampling instance instance-name family inet output]` hierarchy level.

In line sampling supports the `version-ipfix` format that uses UDP as the transport protocol. To configure in line sampling, include the `version-ipfix` statement at the `[edit forwarding-options sampling instance instance-name family inet output flow-server address]` hierarchy level and at the `[edit services flow-monitoring]` hierarchy level.

The following operational commands include in line fpc keywords to display in line configuration information.

- *show services accounting errors*
- *show services accounting flow*
- *show services accounting status*

The Juniper Web Site lists all features that were added to JUNOS Release 10.2.

47.3.1 Configure sFlow Features from the CLI

You configure sFlow technology, designed to monitor high speed switched or routed networks, to continuously monitor traffic at wire speed on all interfaces simultaneously.

1. Enter the following command to configure the IP address of the SevOne NMS appliance.
   [edit protocols sflow]

   ```
   user@switch# set collector <SevOne-IP>
   ```

2. Enter the following command to configure the UDP port on the collector. The default UDP port on SevOne NMS is 6343.

   ```
   [edit protocols sflow]
   [edit protocols sflow] 6343
   ```

3. Enable sFlow technology on a specific interface.
   [edit protocols sflow]

   ```
   user@switch# set interfaces interface-name
   ```

> You cannot enable sFlow technology on a Layer 3 VLAN-tagged interface. You cannot enable sFlow technology on a LAG interface. sFlow technology can be enabled on the member interfaces of the LAG.

4. Enter the following command to specify how often the sFlow agent polls the interface.
[edit protocols sflow]

```
user@switch# set polling-interval seconds
```

Enter 0 (zero) to not poll the interface.

5. Enter the following command to specify the rate at which to sample packets.
[edit protocols sflow]

```
user@switch# set sample-rate number
```

6. You can also configure the polling interval and sample rate at the interface level. The interface level configuration overrides the global configuration.
[edit protocols sflow interfaces]

```
user@switch# set polling-interval seconds
```

[edit protocols sflow interfaces]

```
user@switch# set sample-rate number
```

47.3.2 Juniper Switch

The following configuration enables sFlow monitoring for all interfaces on a Juniper EX switch, sampling packets at 1-in-500, polling counters every 30 seconds and sending the sFlow to SevOne NMS <SevOne-IP> on UDP port 6343.

```
protocols {
sflow {
polling-interval 30;
sample-rate 500;
collector <SevOne-IP> {
udp-port 6343;
}
interfaces ge-0/0/0.0;
interfaces ge-0/0/1.0;
```

47.4 Alcatel

When you enable cflowd on an Alcatel service interface, cflowd collects routed traffic flow samples through a router for analysis. Cflowd is supported on IES and VPRN services interfaces. Layer 2 traffic is excluded. All packets forwarded by the interface are analyzed according to the cflowd configuration. On the interface level, cflowd can be associated with a filter (ACL) or an IP interface.

When you enable cflowd on an interface, all packets forwarded by the interface are subject to analysis according to the global cflowd configuration.

When you configure the cflowd interface option in the `config>router>interface` context, the following requirements must be met to enable traffic sampling on the specific interface.

- Enable cflowd
- Select the `interface>cflowd` interface option
- To omit certain types of traffic from being sampled when the interface sampling is enabled, you can enable the `config>filter>ip-filter>entry>interface-disable-sample` option via an ip-filter or ipv6-filter. You must apply the filter to the service or network interface on which the traffic to be omitted is to ingress the system.

47.4.1 Specify cflowd Options on an IP Interface

Enter the following command.

```
Interface Configurations
CLI Syntax: config>router>if#
cflowd {acl|interface}
no cflowd
```

Depending on the option selected, either `acl` or `interface`, cflowd extracts traffic flow samples from an IP filter or an interface for analysis. All packets forwarded by the interface are analyzed according to the cflowd configuration.

Enable the acl option to enable traffic sampling on an IP filter. You must enable Cflowd (filter-sample) in at least one IP filter entry. Select the interface option to enable traffic sampling on an interface. If cflowd is not enabled (no cflowd) then traffic sampling does not occur on the interface.

> The example below includes the use-vrtr-if-index command. You can use this command to export flow data using interface indexes (ifIndex) instead of using the Alcatel internal global IF index IDs.

```
    Service Interfaces
    CLI Syntax: config>service>vpls service-id# interface ip-int-name
    cflowd {acl|interface}
    active-timeout 20
    inactive-timeout 10
    overflow 10
    rate 100
    use-vrtr-if-index
    collector <SevOne-IP>:9996 version 8
       aggregation
          as-matrix
          raw
       exit
       description <SevOne NMS>
    exit
    collector <SevOne-IP>:9996 version 8
       aggregation
          protocol-port
          source-destination-prefix
       exit
    autonomous-system-type peer
    description "Neighbor collector"
exit
```

47.5 Troubleshoot Flow

SevOne NMS supports most flow formats.

47.5.1 Check for Traffic

If flow data does not display for the device, confirm that SevOne NMS actually receives the data via tcpdump.

Log in to the box and run one of the following commands.

1. Enter the following command to show all incoming flow traffic to SevOne NMS.

   ```
   $ tcpdump -i eth0 port 9996
   ```

2. Enter the following command to show only flow traffic from a specific IP address.

```
$ tcpdump -i eth0 port 9996 | grep '<ip address in question>'
```

3. If data comes into SevOne NMS, you should eventually see a message similar to the following:

   Example

   17:55:47.934113 IP <ip address question>.49359 > \ <SevOne>.9996: UDP, length 1464

4. If no data comes in from the IP address, there may be a routing issue.

47.5.2 Check the Version

If flow data comes in, but nothing displays, the version may be wrong.

Enter the following command to dump the first portions of the packets to the page.

```
$ tcpdump -XX -i eth0 port 9996
```

Something similar to the following should display.

Example

19:55:26.326485 IP <source>.52292 > <destination>.9996: UDP, length 1416
0x0000: 0030 482d 9e1b 0011 5d24 aec0 0800 4500 .0H-....]$....E.
0x0010: 05a4 f187 0000 fb11 ce64 0aff ff0c cc1b ....d......
0x0020: 2435 cc44 270c 0590 2b9f 0005 001d cada $5.Dâ...+.....
0x0030: 5584 45a3 f32e 0cd7 dd44 8682 7d8d 0001 U.E.....D..}...
0x0040: 0000 aa94 ....
19:55:26.326609 IP <source>.58101 > <destination>.9996: UDP, length 1428
0x0000: 0030 482d 9e1b 0011 5d24 aec0 0800 4500 .0H-....]$....E.
0x0010: 05b0 b449 0000 f611 810a 0a00 8f98 cc1b ...I............
0x0020: 2435 e2f5 270c 059c fa38 0007 001b cdb8 $5..â....8......
0x0030: 2bdc 45a3 f32e 11a3 844e 29a1 03b4 0000 +.E.....N).....
0x0040: 0000 0a33

In the example above, the first traffic is v5 and the second is v7 as indicated in the third row's sixth column. The last two digits in the column are the version.

The following is a visual aid to help find the version as indicated by the XX.

Example

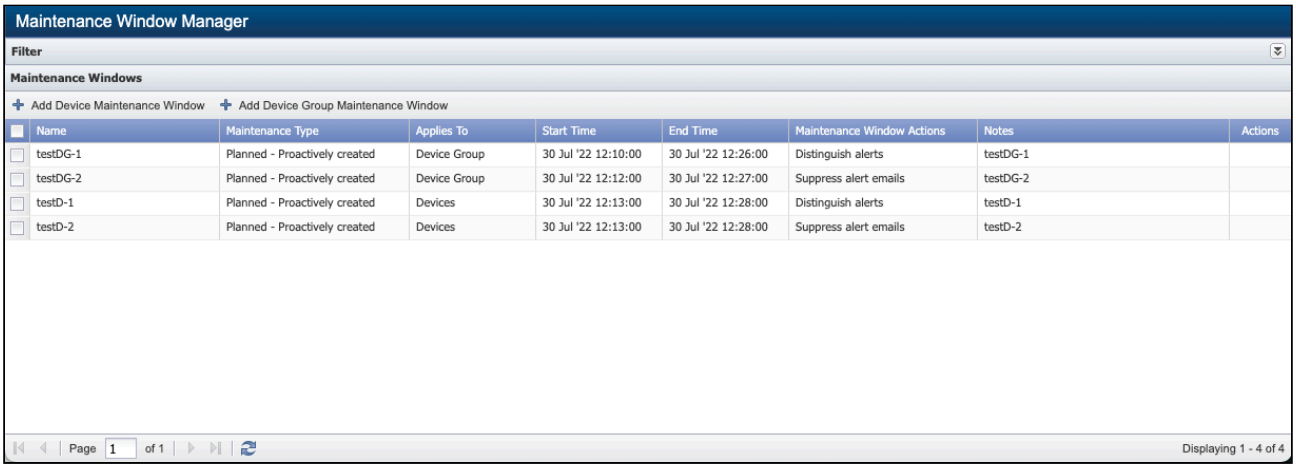
-::.----- IP <source>.----- > <destination>.9996: UDP, length ----
0x0000: ---- ---- ---- ---- ---- ---- ---- ---- ----------------
0x0010: ---- ---- ---- ---- ---- ---- ---- ---- ----------------
0x0020: ---- ---- ---- ---- ---- --XX ---- ---- ----------------
0x0030: ---- ---- ---- ---- ---- ---- ---- ---- ----------------
0x0040: ---- ----

48 Maintenance Windows

The Maintenance Window Manager enables you to view, create, edit, and delete proactive and retroactive device-level maintenance windows.

To access the Maintenance Window Manager from the navigation bar, click the **Administration** menu and select **Maintenance Windows**.

> In order to use the Maintenance Window Manager, you will need to have the page permission **Maintenance Window Configuration** enabled (go to **Administration** -> **Access Configuration** -> **User Role Manager**).



The following information is available for all maintenance windows. To view information for completed maintenance windows or other time ranges, use the Filter panel (please refer to section Apply a Filter below). You can sort on the **Name**, **Start Time**, End **Time**, and **Notes** columns.

- **Name** - The name you give to a maintenance window when you configure it.
- **Maintenance Type** - The type of maintenance being performed.
- **Applies To** - What the maintenance applies to (for example, Devices).
- **Start Time** - The maintenance window start time.
- **End Time** - The maintenance window end time.
- **Maintenance Window Actions** - The action(s) that the maintenance window performs.
- **Notes** - Any additional information that you add when configuring the maintenance window.
- **Actions** - Select  to edit a maintenance window or  delete a maintenance window. You can also right-click on a row and choose the option Edit, to edit the maintenance window or Delete, to delete the maintenance window.

  > When you delete a maintenance window, it will be permanently removed from the system. All functions referencing the maintenance window, including overlays on graphs, will also be removed from the system.

You can also add a UUID column to view the UUIDs for maintenance windows. To do so, perform the following steps:

1. Hover over any of the existing columns and click  .
2. Select **Columns**.
3. Select the check box for **UUID**.

48.1 Create/Edit Maintenance Windows

Perform the following steps to create or edit a maintenance window. The only required fields for a maintenance window are **Name**, **Start Time**, and **End Time**.

48.1.1 Device Maintenance Window

Allows you to select from a list of available devices.

1. Click **Add Device Maintenance Window** to display the pop-up to create a new maintenance window for one or more devices.
   To edit an existing maintenance window, click  under **Actions**. For a maintenance window that is already in progress, you can edit only the **Name**, **Note**, and **End Time** fields.
2. In the **Name** field, enter a name for the maintenance window.
3. In the **Note** field, enter any additional information that you would like to include.
4. Click the **Start Time** field and select a start date and time for the maintenance window. Click **Save**. If you specify a start time in the past, the Actions options below will be unavailable.
5. Click the **End Time** field and select an end date and time for the maintenance window. The maintenance window must last at least three minutes. Click **Save**.
6. Click the **Devices** drop-down and select one or more devices to apply the maintenance window to.
7. Next to **Actions**, select the check box for one or more of the following options. These options are disabled for retroactive maintenance windows.
   - **Suppress alert emails, traps, and webhooks during the maintenance window** - to trigger alerts without sending email notifications or traps or webhook messages.
   - **Distinguish alerts within the maintenance window** - to tag alerts in a maintenance window and cap the Severity level at **Info**. Tagged alerts are used to distinguish between normal alerts and maintenance alerts in the SevOne NMS Alert Summary and in Alerts reports. Tagged alerts will include the prefix **Maintenance Window** in their name. This option is selected by default.
   - **Exclude data from TopN and Group Metrics aggregations during the maintenance window** - to exclude data during the maintenance window from TopN and Group Metrics aggregation calculations.
   - **Exclude data from baselines during the maintenance window** - to exclude data during the maintenance window from baseline calculations.
8. Click **Create** to create a new maintenance window or **Save** to save changes to an existing maintenance window.

48.1.2 Device Group Maintenance Window

Allows you to select a device group with contains one or more devices. When an instance of Device Group maintenance window becomes active, it converts to Device maintenance window and lists all the devices that are contained in the device group selected.

1. Click **Add Device Group Maintenance Window** to display the pop-up to create a new maintenance window for a device group.
   To edit an existing maintenance window, click  under **Actions**. For a maintenance window that is already in progress, you can edit only the **Name**, **Note**, and **End Time** fields.
2. In the **Name** field, enter a name for the maintenance window.
3. In the **Note** field, enter any additional information that you would like to include.
4. Click the **Start Time** field and select a start date and time for the maintenance window. Click **Save**. If you specify a start time in the past, the Actions options below will be unavailable.
5. Click the **End Time** field and select an end date and time for the maintenance window. The maintenance window must last at least three minutes. Click **Save**.
6. Click the **Device Group** drop-down and select a device group to apply the maintenance window to. Only one device group can be selected.

   > When an instance of the maintenance window becomes active, field Device Group converts to Devices and it lists all the devices that belong to the chosen Device Group.
   > If the membership of the Device Group changes while the instance of the maintenance window is active, the change will not impact the instance already in progress (active). A snapshot of Device Group membership at the start time is maintained until the end time. Dynamic changes to the Device Group membership when maintenance window is active, can be rectified ad-hoc by administrators using the *retroactive* maintenance windows feature.

7. Next to **Actions**, select the check box for one or more of the following options. These options are disabled for retroactive maintenance windows.
   - **Suppress alert emails, traps, and webhooks during the maintenance window** - to trigger alerts without sending email notifications or traps or webhook messages.
   - **Distinguish alerts within the maintenance window** - to tag alerts in a maintenance window and cap the Severity level at **Info**. Tagged alerts are used to distinguish between normal alerts and maintenance alerts in the SevOne NMS Alert Summary and in Alerts reports. Tagged alerts will include the prefix **Maintenance Window** in their name. This option is selected by default.
   - **Exclude data from TopN and Group Metrics aggregations during the maintenance window** - to exclude data during the maintenance window from TopN and Group Metrics aggregation calculations.
   - **Exclude data from baselines during the maintenance window** - to exclude data during the maintenance window from baseline calculations.

8. Click **Create** to create a new maintenance window or **Save** to save changes to an existing maintenance window.

48.2 Apply a Filter

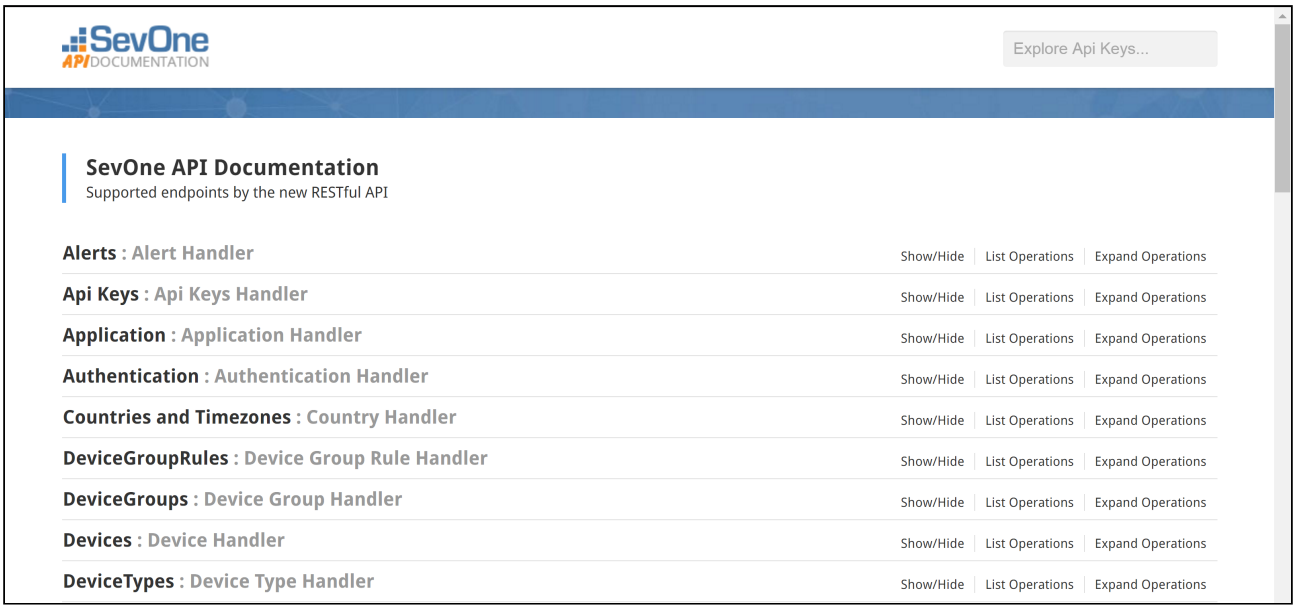
Perform the following steps to filter which maintenance windows display in the Maintenance Windows section.

> Filtering applies to Devices only and cannot be done on Device Groups.

1. In the Filter section, click  to display filter options.
2. Under **Search**, click the **Devices** drop-down and select one or more devices to display maintenance windows for.
3. Under **Time**, select one of the following options:
   - **Active & future windows** - to display active maintenance windows and future maintenance windows.
   - **Active windows** - to display only active maintenance windows.
   - **Future windows** - to display only future maintenance windows.
   - **Completed windows** - to display only completed maintenance windows.
   - **Start time between** Beginning of range **and** End of range - to display maintenance windows within the range you specify. When you select this option, a calendar pop-up appears, enabling you to specify the start date and time as well as the end date and time.
4. Click **Apply Filter** to apply the filter.

48.3 Configure Maintenance Windows through REST API

In order to use SevOne's RESTful API, you must have a valid account in SevOne NMS.



48.3.1 Access Maintenance Windows Endpoints

1. Perform the following actions to sign in:
   a. Go to **http://<PAS hostname or IP address>/api/docs/** (for example, **http://10.129.12.61/api/docs/**).
   b. Click on **Authentication** to view Authentication operations.
   c. Under **Authentication**, click on **POST**.
   d. Under **Parameters**, all the way to the right, locate the **Model Schema** field. Click on the field to copy its content to the **user** field.
   e. On the left side of the **Parameters** section, locate the **user** field. After **"name":**, replace **string** with a SevOne NMS user name. Make sure to enter it within the quotes.
   f. After **"password":**, replace **string** with the corresponding SevOne NMS password. Make sure to enter it within the quotes.
   g. At the bottom of the **POST** section, click the **Try it out!** button.

h. Scroll down to the **Response Body** field. You should see a long alphanumeric string after **<token>**. This is the token that you need. Double-click the token to select it. Then copy it.
i. In the upper right corner of the **SevOne API Documentation** page, locate the **Explore Api Keys...** field. Paste the token into this field. You should now have permissions to perform operations.

2. Click on **Maintenance Windows** to view Maintenance Window Handler operations.
3. Continue to the Operations section to perform specific operations related to maintenance windows.

48.3.2 Operations

The available operations allow you to create, update, and delete maintenance windows. You can also view information about existing maintenance windows. A description of each operation appears on the right side of the page. Additional documentation appears for each item below when you click the **Model** tab under **Response Class (Status 200)**.

48.3.2.1 Create Maintenance Windows

Perform the following actions to create maintenance windows for devices.

1. Click on **POST /api/v1/maintenancewindows** to create a maintenance window for one or more devices.
2. On the right side of the page, click on the **Model Schema** field to copy its content to the **maintenanceWindowDto** field.
3. The following settings can be configured:
    - **actions** - The action(s) to apply to the maintenance window. Options include the following:

      When creating a maintenance window with a start date in the past, do not provide input for any of the actions below. Actions are not available for retroactive maintenance windows.

        - **SUPPRESS_ALERT_NOTIFICATIONS** - to trigger alerts without sending traps or email notifications or webhook messages.
        - **CATEGORIZE_ALERTS** - to tag alerts in a maintenance window and cap the Severity level at **Info**. Tagged alerts are used to distinguish between normal alerts and maintenance alerts in the SevOne NMS Alert Summary and in Alerts reports. Tagged alerts will include the prefix **Maintenance Window** in their name.
        - **EXCLUDE_DATA_FROM_AGGREGATION** - to exclude data during the maintenance window from TopN and Group Metrics aggregation calculations.
        - **EXCLUDE_DATA_FROM_BASELINES** - to exclude data during the maintenance window from baseline calculations.
    - **deviceIds** - The device(s) that the maintenance window applies to. Provide one or more device IDs and separate device IDs using a comma. Use the REST API to get device IDs for the devices that the maintenance window applies to.
    - **maintenanceType** - The type of maintenance. Currently the only option here is **PLANNED**.
    - **name** - The name you give to the maintenance window. The default is set to **string**, which means you will need to give the maintenance window a name. Otherwise, the name will appear as **string** in the SevOne NMS Maintenance Window Manager.
    - **notes** - Additional information that you would like to include. To leave this blank, delete **string** from the line **"notes": "string"**. Otherwise, the text **string** will apper under **Notes** in the SevOne NMS Maintenance Window Manager.
    - **scheduleInstance** - The time range of the maintenance window. The default format is UNIX timestamp in milliseconds (for example, 1498177530000). Times can also be expressed using ISO 8601 format. You can specify the format in the **Date-Format** header (under **Parameters**). The specified time must be UTC, and the maintenance window must last at least three minutes.
        - **beginDateTime** - The date and time that the maintenance window should start.
        - **endDateTime** - The date and time that the maintenance window should end. The maintenance window must last at least three minutes.
4. After configuring the maintenance window, click **Try it out!** at the bottom of the section.

48.3.2.2 View Maintenance Windows

Perform the steps below to view information for maintenance windows.

View All Maintenance Windows

1. Click on **GET /api/v1/maintenancewindows**.
2. At the bottom of the section, click **Try it out!**.

3. See the **Response Body** field for information about existing maintenance windows. **totalElements** indicates the total number of maintenance windows. Scroll down to view specific information about each maintenance window, including the maintenance window ID.

View a Maintenance Window Using an ID

1. You will need the ID for the maintenance window that you want to view information for. You can get IDs for existing maintenance windows by performing the steps above (**View All Maintenance Windows**). Copy the ID for the maintenance window you would like to view (for example, **20f3db94-9577-4ceb-92cd-b988d66fcaaf**).
2. Click on **GET /api/v1/maintenancewindows/{id}**.
3. Under **Parameter**, in the **id** field, paste the ID for the maintenance window you would like to view information for.
4. Click **Try it out!**.
5. See the **Response Body** field for information about that maintenance window.

View Maintenance Windows Using a Filter

1. Click on **POST /api/v1/maintenancewindows/filter**.
2. On the right side of the page, click on the **Model Schema** field to copy its content to the **filter** field.
3. Provide input for **actions**, **deviceIds**, etc., depending on how you would like to filter results.

   Filter parameters support inexact matches:
   - **name** - will match any maintenance window containing the specified substring.
   - **actions** - will match a maintenance window if it contains any of the specified actions.
   - **beginDateTime**/**endDateTime** - will match any maintenance window overlapping the specified period.
   - **deviceIds** - will match a maintenance window if it contains any of the specified devices.
4. Click **Try it out!**.
5. See the **Response Body** field for results.

48.3.2.3 Edit Maintenance Windows

1. You will need the ID for the maintenance window that you want to edit. You can get IDs for existing maintenance windows by performing the steps above (**View All Maintenance Windows**). Copy the ID for the maintenance window you would like to edit (for example, **20f3db94-9577-4ceb-92cd-b988d66fcaaf**). For a maintenance window that is already in progress, you can edit only the **name**, **notes**, and **endDateTime** fields. If the maintenance window has already ended, you can edit only the **name** and **notes** fields.
2. Click on **PUT /api/v1/maintenancewindows/{id}**.
3. Under **Parameter**, in the **id** field, paste the ID for the maintenance window you would like to edit.
4. On the right side of the page, click on the **Model Schema** field to copy its content to the **maintenanceWindowDto** field.
5. In the **maintenanceWindowDto** field, provide input for any information you would like to modify. Make sure to provide input for all fields, including fields that you aren't modifying. If you omit any fields, the data for those fields will be deleted.
6. Click **Try it out!**.

48.3.2.4 Delete Maintenance Windows

1. You will need the ID for the maintenance window that you want to delete. You can get IDs for existing maintenance windows by performing the steps above (**View All Maintenance Windows**). Copy the ID for the maintenance window you would like to delete (for example, **20f3db94-9577-4ceb-92cd-b988d66fcaaf**).
2. Click on **DELETE /api/v1/maintenancewindows/{id}**.
3. Under **Parameter**, in the **id** field, paste the ID for the maintenance window you would like to delete.
4. Click **Try it out!**.

48.4 Alert Scenarios

Alert processing does not stop during a scheduled maintenance window when you select the check box for **Distinguish alerts within the maintenance window**. Instead, alerts that trigger during a scheduled maintenance window are annotated differently than normal alerts. Whether an alert is or is not classified as occurring within a maintenance window is based on the timestamp indicating when the alert conditions trigger. This classification is not based on timestamps associated with collected data samples. The difference between the timestamp of the final collected sample that triggered the alert and the evaluation time of the alert should be small (within one or two poll intervals). This behavior is consistent with the check box option **Suppress alert emails, traps, and webhooks during the maintenance window**.

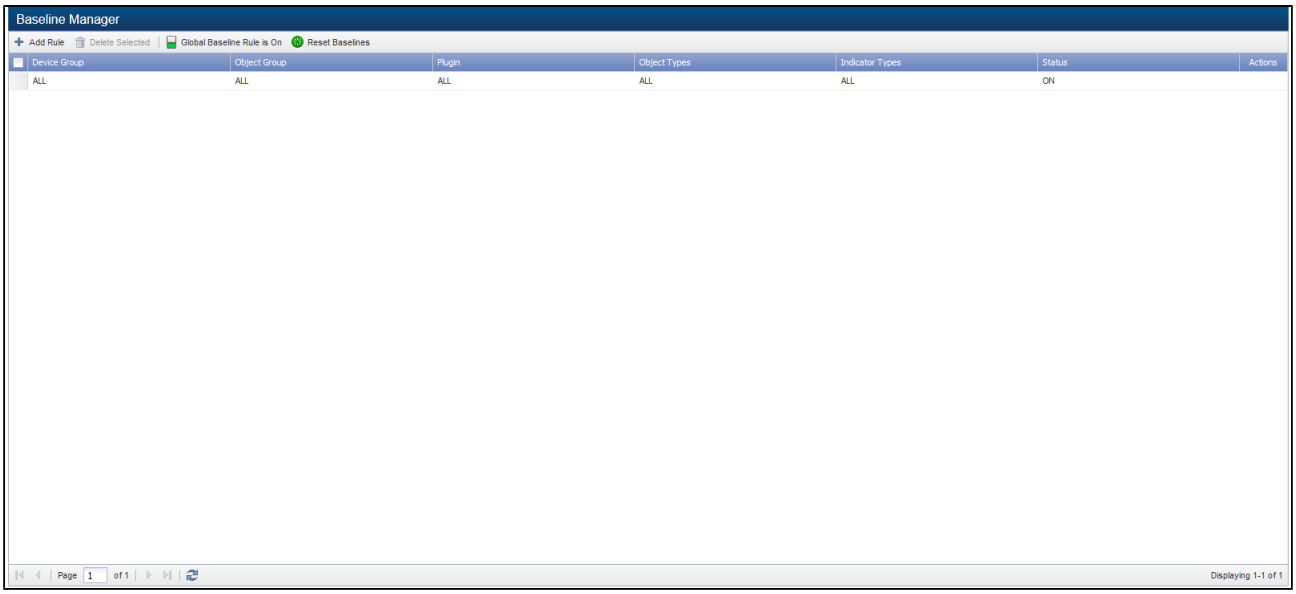
The following table describes the expected system behavior for different alert scenarios, based on the timing of the alert trigger.

Alert Trigger Timing Relative to Maintenance Window	System Behavior
Alert triggers before a maintenance window and stops triggering before a maintenance window.	• Email / Trap notifications / webhook messages are sent based on **policy** / **threshold** configuration.
Alert triggers after a maintenance window ends.	• Email / Trap notifications / webhook messages are sent based on **policy** / **threshold** configuration.
Alert triggers before a maintenance window begins and stops triggering during a maintenance window.	• Email / Trap notifications / webhook messages are sent based on the **policy** / **threshold** configuration before the maintenance window. • Email / Trap notifications / webhook messages will be suppressed during the maintenance window if **Suppress alert emails, traps, and webhooks during the maintenance window** option is checked.
Alert triggers during a maintenance window and stops triggering after a maintenance window.	• Email / Trap notifications / webhook messages will be suppressed during the maintenance window if **Suppress alert emails, traps, and webhooks during the maintenance window** option is checked. • Email / Trap notifications / webhook messages will be sent after the maintenance window passes.
Alert triggers during a maintenance window and stops triggering within the same maintenance window.	• No email / trap notifications / webhook messages will be sent if **Suppress alert emails, traps, and webhooks during the maintenance window** option is checked.
Alert triggers prior to a maintenance window, continues to trigger during the maintenance window, and stops triggering after the maintenance window has ended.	• No email / trap notifications / webhook messages will be sent during the maintenance window if **Suppress alert emails, traps, and webhooks during the maintenance window** option is checked. • Email / Trap notifications / webhook messages will be sent **before** / **after** the maintenance window.

49 Baseline Manager

The Baseline Manager enables you to define rules for when to create baselines and enables you to reset a time frame within a stored baseline to eliminate unnatural dips and spikes. By default, SevOne NMS creates a baseline for every polled indicator. The Baseline Manager enables you to define rules to prevent the creation of baselines you deem irrelevant.

To access the Baseline Manager from the navigation bar, click the **Administration** menu and select **Baseline Manager**.



The Cluster Manager > **Cluster Settings** tab enables you to change the baseline granularity. Default baseline granularity is 900 seconds (15 minutes). For the default baseline granularity, SevOne NMS collects data for 15 minutes and stores the 15 minutes of data in a bucket. The data in each bucket is averaged to create one data point per bucket. The baseline for each indicator uses one week of data for a total of 672 data points per baseline for each indicator. A new indicator starts to create a baseline upon first poll but thresholds require a full week of poll data before they can trigger a baseline based alert. As time passes, baselines become a more accurate representation of the indicator's average operation.

49.1 Baseline Rules

The Global Baseline rule appears first in the list and you cannot edit or delete the Global Baseline rule. The Global Baseline rule enables you to create baselines for all indicators or to not create baselines for all indicators. The Global Baseline rule is on by default to create baselines for all indicators.

You can leave the Global Baseline rule on and define rules for the indicator types from which you do not want to create baselines or you can turn the Global Baseline rule off and define rules to create only the baselines that matter to you.

> The Global baseline rule determines whether or not SevOne NMS creates baseline data across your entire network. If you turn this rule off, you stop the creation of baselines and delete all baseline data for every indicator type that does not have a specific baseline creation rule.

Click **Add Rule** to display a pop-up that enables you to define baseline rules. All fields are optional and each selection provides more granularity.

1. Click the **Device Group** drop-down and select the device group/device type that contains the object for which to create a baseline rule.
2. Click the **Object Group** drop-down and select the object group that contains the objects for which to create a baseline rule.
3. Click the **Plugin** drop-down and select the plugin that polls the object for which to create a baseline rule.
4. When available, click the **Object Type** drop-down and select the object type for which to create a baseline rule.
5. When available, click the **Indicator Type** drop-down and select the indicator type that contains the indicator for which to create a baseline rule.
6. Select one of the **Create Baselines** options.
    - Select **On** to create a baseline for data that meet the rule criteria.
    - Select **Off** to not create a baseline for data that meets the rule criteria.

7. Click **Add**.

49.1.1 Rule Examples

Define a rule to not create baselines for all Linux devices and leave the Global Baseline Rule set to on.

Device Group	Object Group	Plugin	Object Types	Indicator Types	Status
ALL	ALL	ALL	ALL	ALL	ON
Linux	ALL	ALL	ALL	ALL	OFF

Define a rule to not create baselines for any device SNMP poller fans and leave the Global Baseline Rule set to on.

Device Group	Object Group	Plugin	Object Types	Indicator Types	Status
ALL	ALL	ALL	ALL	ALL	ON
Generic	ALL	SNMP Poller	Fan Condition	Fan Status	OFF

Define rules to not create baselines for all router SNMP Poller information except for In Octets and Out Octets indicators and leave the Global Baseline Rule set to on. This requires three rules.

Device Group	Object Group	Plugin	Object Types	Indicator Types	Status
ALL	ALL	ALL	ALL	ALL	ON
Router	ALL	SNMP Poller	Interface	ALL	OFF
Router	ALL	SNMP Poller	Interface	HC Out Octets	ON
Router	ALL	SNMP Poller	Interface	HC In Octets	ON

49.2 Reset Baselines

When a baseline contains a time frame that includes an unnatural dip or spike, perform the following steps to reset the baseline. The Cluster Manager > **Cluster Settings** tab enables you to define the baseline granularity. The Reset Baselines pop-up displays a message "Baselines are <n> minutes long" to inform you of the baseline granularity. You cannot reset a baseline time frame that is less than the length of the baseline granularity.

1. Click **Reset Baselines** to display the Reset Baselines pop-up.
2. Click the **Reset Type** drop-down. The fields that follow are dependent on the selection you make here.
    - Select **Device** to reset the baseline for a specific device and perform the following steps.
        i. Click the **Device** drop-down and select the device that contains the indicator for which to reset a baseline.
        ii. Click the **Plugin** drop-down and select the plugin that polls the object.
        iii. Click the **Object** drop-down and select the object that contains the indicator.
        iv. Click the **Indicator** drop-down and select the indicator for which to reset the baseline.
    - Select **Device Group** to reset the baseline for a device group/device type and perform the following steps.
        i. Click the **Device Group** drop-down and select the device group/device type that contains the object type for which to reset a baseline.
        ii. Click the **Plugin** drop-down and select the plugin that polls the object type.
        iii. Click the **Object Type** drop-down and select the object type that contains the indicator type.
        iv. Click the **Indicator Type** drop-down and select the indicator type for which to reset the baseline.
    - Select **Object Group** to reset the baseline for an object group and perform the following steps.
        i. Click the **Object Group** drop-down and select the object group that contains the object for which to reset a baseline.
        ii. Click the **Plugin** drop-down and select the plugin that polls the object type.
        iii. Click the **Object Type** drop-down and select the object type that contains the indicator type.
        iv. Click the **Indicator Type** drop-down and select the indicator type for which to reset the baseline.
3. Complete the fields in the Time Frame section to define the beginning of the time frame to reset and the end of the time frame. You must enter the time frame in Coordinated Universal Time (UTC) which is also known as Greenwich Mean Time (GMT).
    - Enter two time values that are equal to reset the baselines for the entire week (Sunday 0:00 to Sunday 0:00).
    - Enter a time frame to reset for the one hour when an event causes a skew in data for one hour (Monday 10:00 to Monday 11:00).
4. Click **Reset Baselines**. You are prompted to acknowledge that you are aware that you cannot undo this should you choose to proceed.

50 SevOne Data Bus

> This topic describes the Command Line Interface (CLI) based configuration of SevOne Data Bus only.
> SevOne Data Bus can be configured using the Graphical User Interface. For details, please refer to **Cluster Manager** > section **SevOne Data Bus Configuration**.

SevOne Data Bus is a SevOne component that listens for new poll points for devices and publishes this data to Apache Kafka broker or Apache Pulsar broker cluster. When poll points are detected, the SevOne REST API is used to enrich the data set with readable names and additional data prior to publishing to Kafka. SevOne Data Bus enables you to perform analyses by combining SevOne data with other types of data–for example IT and Business data. It allows you to stream real-time data from SevOne NMS to an external message bus. Any application capable of integrating with Apache Kafka can subscribe to the data published.

To use SevOne Data Bus, you will need to purchase the license for it. For more information, talk to your Technical Account Manager (if applicable) or SevOne's Sales Engineering team.

> SevOne Data Bus uses Kafka Client library version 3.4.0 for publishing data to the Kafka broker. Please visit https://kafka.apache.org/32/documentation.html to determine compatibility of SevOne Data Bus with Kafka broker versions.

50.1 Configure, Start, and Stop SevOne Data Bus

50.1.1 Configure and Start SevOne Data Bus

Perform the following steps to configure and start SevOne Data Bus. You will need to do this for each peer in your cluster. Once you configure the **application.conf** file on the first peer, simply copy the file to each additional peer in the cluster.

> You will need to have a license for SevOne Data Bus in order to enable it. Without a valid license, the process will exit.

> SevOne Data Bus can be configured for devices/objects polled at 1-minute frequency as long as indicators/sec are within SevOne NMS polling specification per appliance size.

1. SSH into your SevOne NMS appliance or cluster. If you SSH in as root, it is not necessary to use **sudo** with commands.

   > If you do not SSH in as **root**, you will need to precede commands with **sudo**.

   > Enable / Disable SevOne Data Bus

   Enable SDB

   ```
   $ SevOne-act data-bus enable
   ```

   Disable SDB

   ```
   $ SevOne-act data-bus disable
   ```

2. Copy application.conf.template in **/etc/sevone/data-bus** to application.conf.

   > If upgrading from an earlier version of SevOne Data Bus, make sure to backup prior the application.conf file before applying the SevOne Data Bus 1.4 template and update accordingly.

3. Using the text editor of your choice, open **application.conf** in **/etc/sevone/data-bus**.

4. For SevOne NMS REST API settings, configure the following under **api**:
    - api
        - authentication
            - **username** - The SevOne NMS user name. This should be for an admin account.
            - **password** - The corresponding SevOne NMS password.
5. For the SevOne Data Bus output format, configure the following under **databus**:
    - databus
        - output
            - **format** - The data output format, which can be either **avro** or **json**.
6. To filter SevOne Data Bus output, configure the information below under **filters**. This is an array of filters, and there are rules for each filter.

    > SevOne Data Bus supports both allowlist and blocklist filtering are supported. There are two separate sections:
    > - **include-filters** for allowlist filter rules
    > - **exclude-filters** for blocklist filter rules

    - filters - The array of filters. Configure the following information for each filter you would like to add.

        > Filtering on device groups and/or object groups with large membership counts can result in timeout errors in the cache.

        - **name** - The name of the filter.
        - **rules** - Configure the following rules for each filter. You can configure all four of the following IDs for any number of rows. A value of **-1** represents any ID. A data point matches a row if it matches all of the IDs in that row. A data point is included when it matches any of the rows. If you have both allowlist and blocklist, and there are overlapping rules in both list, then the blocklist will take the dominant position. If a row has **-1** for all IDs, then nothing will be excluded. You can get IDs for the items listed below using the SevOne NMS REST API. For more information, see the SevOne NMS *RESTful API Quick Start Guide*.
            - **devGrpID** - The device group ID.
            - **objGrpID** - The object group ID.
            - **devID** - The device ID.
            - **objID** - The object ID.

Examples of rules

```
# Filters for the Data Bus output
# Note: SDB only exports data from the local peer, so these filters only apply to
local data.
filters = [
  {
    name = "default"
    # Specify your filters as different elements in this array
    # by specifying an ID that you would like to be included.
    # A value of -1 is interpreted as any ID.

    # Each column in a given filter is combined with logical AND.
    # Multiple filters are combined with logical OR.
    rules = [
      # Example: Include everything
      {devGrpID = -1, objGrpID = -1, devID = -1, objID = -1}

      # Example: Include only devices 5 and 6
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = -1},
      # {devGrpID = -1, objGrpID = -1, devID = 6, objID = -1}

      # Example: Include device 5 only if it's in device group 2 and device 6
      # {devGrpID = 2, objGrpID = -1, devID = 5, objID = -1},
      # {devGrpID = -1, objGrpID = -1, devID = 6, objID = -1}

      # Example: Include only objects 2, 3, and 4 from device 5
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 2},
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 3},
```

```
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 4}
    ]
  }
]

exclude-filters = [
  {
    name = "block-list"
    # Specify your filters as different elements in this array
    # by specifying an ID that you would like to be included.
    # A value of -1 is interpreted as any ID.

    # Each column in a given filter is combined with logical AND.
    # Multiple filters are combined with logical OR.
    rules = [
      # Example: Exclude everything
      {devGrpID = -1, objGrpID = -1, devID = -1, objID = -1}

      # Example: Include only devices 5 and 6
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = -1},
      # {devGrpID = -1, objGrpID = -1, devID = 6, objID = -1}

      # Example: Exclude device 5 only if it's in device group 2 and device 6
      # {devGrpID = 2, objGrpID = -1, devID = 5, objID = -1},
      # {devGrpID = -1, objGrpID = -1, devID = 6, objID = -1}

      # Example: Exclude only objects 2, 3, and 4 from device 5
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 2},
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 3},
      # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 4}
    ]
  }
]
```

7. You can configure multiple Kafka brokers or Pulsar brokers, exclusively, as output destinations. Under **output** -> **publishers**, configure the settings below for each publisher you would like to add. Default producer configurations shared by all publishers are also included here (**output** -> **default** -> **producer**).
    - output
        - default

> **Applies to Kafka or Pulsar**
>
> The default section contains producer configurations shared by all publishers. These default configurations are overwritten by the settings you specify for each publisher. You can modify the following default producer settings or modify/add any other valid Kafka or Pulsar producer configuration in addition to the ones below.

            - producer - A list of the Kafka producer configuration settings.
                - **acks** - The number of acknowledgements that the leader must receive before a request is considered complete. Options include **0**, **1**, and **-1**. -1 will wait until all of the replicas (cluster members that are replicating the partition) acknowledge the message. For this reason, -1 is considered to be the most robust, albeit slowest, option.
                - **retries** - The number of times to retry sending a failed message.
                - **linger.ms** - The amount of time in milliseconds for messages to remain in the producer queue before message batches are created.
                - **batch.size** - The number of messages batched into a MessageSet.
                - **request.timeout.ms** - The amount of time in milliseconds that the client will wait for a request response.
                - **max.in.flight.requests.per.connection** - The maximum number of unacknowledged requests sent to a broker.
            - pulsar-producer - A list of Pulsar producer configuration settings.
                - **batchingMaxMessages** - The maximum number of batched message for Pulsar producer.
                - **sendTimeoutMs** - The number in milliseconds for which Pulsar will wait to report an error if a message is not acknowledged by the server.
        - publishers - The array of publishers. Configure the following information for each publisher that you would like to add.

- Common Settings (for Kafka or Pulsar)
    - **name** - The name of the default publisher.
    - **filters** - The array of filters to include for the publisher. Filters should be listed within **[...]**. Combine results from multiple filters using logical **OR**.
    - **topic** - The Kafka or Pulsar topic that SevOne Data Bus writes to. This can be anything you want. For example, "sdb".
    - **isLive** - The current publisher accepts historical data only when set to false.
    - **type** - The type of this publisher. If not set, SevOne Data Bus will set the default value as **kafka** for this field. To export to Pulsar, user **must explicitly** set this field to **pulsar**.
- Kafka-specific Settings
    - **producer** - Configure the following information for the producer. Please refer to https://kafka.apache.org/25/documentation.html#producerconfigs for available settings.
        - **bootstrap.servers** - The Kafka server, also known as a *broker*, and the port number for accessing the server. Use the format **{server IP address or hostname}:{port number}**. For example, "10.129.13.10:9092".
        - **security.protocol** - Protocol used to communicate with the Kafka broker. For example, **SSL**.
        - **ssl.truststore.location** - The location of the trust store file.
        - **ssl.truststore.password** - The password of the trust store file. If a password is not set, access to the trust store is still available but integrity checking is disabled.
        - **ssl.protocol** - The SSL protocol used to generate the SSLContext. The default setting is **TLSv1.2** in most cases. The allowed values in recent JVMs are **TLSv1.2** and **TLSv1.3**.
        - **ssl.keystore.location** - The location of the key store file. This is optional for client and can be used for two-way authentication for client.
        - **ssl.keystore.password** - The store password for the key store file. This is optional for client and only needed if **ssl.keystore.location** is configured.
        - **ssl.key.password** - The password of the private key in the key store file. This is optional for a client.
        - **ssl.endpoint.identification.algorithm** - This setting is only needed to turn off hostname validation. Hostname validation is on by default and turned off when this property is set to an empty string.
- Pulsar-specific Settings
    - **client** - Configure the following for Pulsar client. Please refer to https://pulsar.apache.org/docs/en/client-libraries-java/ for available settings.
        - **serviceUrl** - The service URL of Pulsar.
        - **connectionTimeoutMs** - Duration to wait for a connection to a broker to be established.
        - **useTls** - Set to **true**. For example, useTls=true.
        - **tlsTrustCertsFilePath** - Set to **"/path/to/ca.cert.pem"**. For example, tlsTrustCertsFilePath="/path/to/ca.cert.pem".
        - **tlsAllowInsecureConnection** - Set to **true**. For example,tlsAllowInsecureConnection=true.
        - **authPluginClassName** - Authorized plugin class name. For example, authPluginClassName="org.apache.pulsar.client.impl.auth.AuthenticationTls"
    - **tenant** - Pulsar service tenant name.
    - **namespace** - Pulsar service namespace.
    - **topic-type** - Can be set to "persistent" or "non-persistent".
    - **producer** - Please see the table below fo Pulsar producer settings.

Pulsar Producer	Description	Values
messageRoutingMode	Set the `MessageRoutingMode` for a partitioned producer. Please refer to the following link for details. • http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/MessageRoutingMode.html	• "SinglePartition" • "RoundRobinPartition"
compressionType	Set the compression type for the producer.	• "NONE" • "LZ4" • "ZLIB" • "ZSTD" • "SNAPPY
autoUpdatePartitions	If enabled, partitioned producer will automatically discover new partitions at runtime.	[Boolean](true)
batchingEnabled	Enable Batching.	[Boolean](false)
batchingMaxMessages	Set the maximum number of messages permitted in a batch.	[Int](1000)
blockIfQueueFull	Set whether the send operations should block when the outgoing message queue is full. Please refer to the following links for details. • http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/Producer.html#send-T- • *http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/Producer.html#sendAsync-T-*	[Boolean](false)
maxPendingMessages	Set the maximum size of the queue holding the messages pending to receive an acknowledgment from the broker.	[Int](500000)

Pulsar Producer	Description	Values
hashingScheme	Change the `HashingScheme` used to chose the partition on where to publish a particular message. Please refer to the following link for details. • http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/HashingScheme.html	• "JavaStringHash" • "Murmur3_32Hash"

The [String] in the table above means that you can put any value of that type in there. The value inside () is the default value for the setting.

8. To monitor SevOne Data Bus data processing, configure the following **http** and/or **https** settings under **status**:
    - status
        - http
            - **enabled** - Whether the http status page is enabled (**true**) or disabled (**false**).
            - **port** - The port that the SevOne Data Bus status page runs on. The default port is **8082**.
        - https
            - **enabled** - Whether the https status page is enabled (**true**) or disabled (**false**).
            - **secure_port** - The secure port that the SevOne Data Bus status page runs on. The default port is **8443**.
            - **private_key_password** - The private key password.
            - **keystore_password** - The keystore password.
            - **keystore_path** - The path to the keystore. The default is **"/etc/sevone/data-bus/sdb.keystore"**.
9. Save and close the file.
10. If you have a cluster, make sure to copy the configured application.conf file to each peer in the cluster (to the directory /etc/sevone/data-bus).
11. Execute one of the following commands to add SevOne Data Bus to the default runlevel. This ensures that SevOne Data Bus will restart after a system reboot. If you do not add SevOne Data Bus to the default runlevel, you will need to manually restart SevOne Data Bus after rebooting.
    - Gentoo SevOne NMS:

```
$ rc-update add sevone-data-bus default
```

    - CentOS SevOne NMS:

```
$ /usr/bin/supervisorctl update
```

12. Execute the following command to adjust the SevOne Data Bus configuration for your PAS size. Replace **{PAS_SIZE}** with **pas300k**, **pas200k**, **pas60k**, **pas40k**, **pas20k**, **pas10k**, or **pas5k**. SevOne Data Bus will start when you run this command.

```
$ SevOne-select data-bus appliance {PAS_SIZE}
```

50.1.1.1 Configure SevOne Data Bus to Start on Reboots

After you have configured the application.conf, perform the following steps to ensure that SevOne Data Bus starts on reboots. This applies only if you have SevOne NMS running on CentOS. This step does not apply if SevOne NMS is running on Gentoo.

If you do not SSH in as **root**, you will need to precede commands with **sudo**.

1. Using the text editor of your choice, open **SevOne-data-bus.ini** in **/etc/supervisord.d/**.
2. In the following line, change **false** to **true**:

```
autostart=false
```

3. Save and close the file.
4. Execute the following command.

```
$ /usr/bin/supervisorctl update
```

50.1.1.2 Restart SevOne Data Bus

Any time you make changes to the SevOne Data Bus configuration after the initial configuration, you will need to restart SevOne Data Bus using one of the the following commands:

Gentoo SevOne NMS:

```
$ /etc/init.d/sevone-data-bus restart
```

CentOS SevOne NMS:

```
$ /usr/bin/supervisorctl restart SevOne-data-bus
```

50.1.2 Optional Configurations

The following configurations are optional.

50.1.2.1 Configure Datapoint Enrichment

SevOne Data Bus outputs the data in either **avro** or **json** formats.

> When using **json**, the output format is fixed.
> When using **avro**, users can configure the **json** schema to customize the fields that SevOne Data Bus exports. **avro** output is controlled by a schema file. The schema is in **json** format and can be found in **/etc/sevone/data-bus/databusmsg.json**.

Using a text editor of your choice, update the schema for the **avro** output in **/etc/sevone/data-bus/databusmsg.json** file, for example, to include/exclude the following supported fields.

```
{
"fields": [
{ "name": "deviceId", "type": "int"},
{ "name": "deviceName", "type": "string"},
{ "name": "deviceIp", "type": "string"},
{ "name": "peerId", "type": "int"},
{ "name": "objectId", "type": "int"},
{ "name": "objectName", "type": "string"},
{ "name": "objectDesc", "type": "string"},
{ "name": "pluginId", "type": "int"},
{ "name": "pluginName", "type": "string"},
{ "name": "indicatorId", "type": "int"},
{ "name": "indicatorName", "type": "string"},
```

```
{ "name": "format", "type": "int"},
{ "name": "value", "type": "string"},
{ "name": "time", "type": "double"},
{ "name": "clusterName", "type": "string"},
{ "name": "peerIp", "type": "string"},
{ "name": "objectType", "type": "string"},
{ "name": "units", "type": "string"}
],
"name": "databusmsg",
"type": "record"
}
```

⚠ Indicator type **units** displays the info in **data units** and not in **display units**.

50.1.2.2 Publish Metrics in Confluent Control Center

To publish metrics in Confluent Control Center, follow the instructions for adding producer/consumer interceptors to the client configuration at https://docs.confluent.io/current/control-center/docs/installation/clients.html#interceptor-installation.

After you have done that, you will need to add the Confluent interceptor classes to the application.conf file. Under **output** -> **default** -> **producer**, add the line below.

- output
  - default
    - **producer** - Add the following line:

```
interceptor.classes =
"io.confluent.monitoring.clients.interceptor.MonitoringProducerInterceptor"
```

50.1.2.3 Validate and Register Schema with Confluent Schema Registry

SevOne Data Bus will validate and register the schema with the Confluent schema registry when it starts. To enable this feature, you will need to add the schema registry server URL to the application.conf file.

Add the URL to the following line under **schema-registry** in the application.conf file. Replace **SCHEMA_REGISTRY_SERVER_HOST** with the server host name or IP address and **SCHEMA_REGISTRY_SERVER_PORT** with the server port.

To enable this, you will need to remove the **#** that precedes **url** at the beginning of the line.

```
url: "http://<SCHEMA_REGISTRY_SERVER_HOST>:<SCHEMA_REGISTRY_SERVER_PORT>"
```

You can also configure the subject name for the schema in the following line. Replace the default subject name, **sevone-sdb**, with your new subject name within the quotes.

```
subject: "sevone-sdb"
```

ⓘ If the compatibility validation fails or an exception is thrown during validation, SevOne Data Bus will terminate.

50.1.2.4 Map Devices to a Specific Kafka Partition

SevOne Data Bus supports Kafka partitions based on the key field. A key is composed of key fields (key-fields) and a key delimiter (key-delimiter). Kafka will handle the message distribution to different partitions based on the key and will ensure that messages with the same key go to the same partition.

Under **kafka**, for **key-fields**, change **["deviceId", "objectId"]** to **["deviceId"]**. This is necessary for mapping devices to a specific partition.

```
key-fields: ["deviceId", "objectId"]
```

50.1.2.5 Encryption & Authentication for Kafka

> ⚠ If the configuration applies globally to all publishers, place the properties in the **output.default.kafka-producer** section of the configuration file. If the configuration is not global, place the properties in the producer section of the appropriate entry in the publishers list.

50.1.2.5.1 SSL / TLS only Encryption

To enable SSL, a certificate is required. This certificate must be part of a Java key store (JKS) file which is referred to as the **truststore**. The *truststore* establishes a chain of trust from the SSL certificate to a *root* certificate.

If the certificate is provided as a Privacy-Enhanced Mail (PEM) or Distinguished Encoding Rules (DER) encoded file, it can be imported into a new JKS file on the command line. Let's assume the certificate is *mycert.pem*, the certificate will be known as *MyCert* in the truststore, the new JKS file will be *mytruststore.jks*, and the password for the JKS file will be *MyJKSPassword*.

Create a new JKS file

```
$ keytool -importcert -keystore mytruststore.jks -alias MyCert -file mycert.pem -
storepass MyJKSPassword
```

The truststore file must include the SSL certificate and any intermediate certificates required to provide a trust relationship back to a trusted root certificate.

If a self-signed root certificate is used, it must be added to the keystore file as well.

> ⚠ Self-signed certificates must only be used in non-production environments.

With the truststore in place, properties must be added to SevOne Data Bus configuration file, **/etc/sevone/data-bus/application.conf**.

If the SSL configuration applies to all defined publishers, properties must be added to the **output.default.kafka-producer** section of the configuration file.

Example

```
...
output {
  ...
  default {
    ...
    kafka-producer {
      ...

      security.protocol=SSL
      ssl.truststore.location=/etc/sevone/data-bus/mytruststore.jks
      ssl.truststore.password=MyJKSPassword
    }
...
```

If the SSL configuration does not apply to all the defined publishers, properties must be added to each publisher to which the configuration applies in the publisher's **producer** section.

Example

```
...
output {
  ...
  publishers = [
    ...
    {
      name="Some Publisher"
      type="kafka"
      ...

      producer {
        security.protocol=SSL
        ssl.truststore.location=/etc/sevone/data-bus/mytruststore.jks
        ssl.truststore.password=MyJKSPassword
        ...
      }
    }
...
```

For advanced scenarios, additional settings are available. Please refer to the document(s) of the product in use.

The *Security* section of **Apache Kafka** documentation can be found in https://kafka.apache.org/documentation/#security
The *Security* section of **Confluent Platform** documentation can be found in https://docs.confluent.io/platform/current/security/general-overview.html

Please **restart** SevOne Data Bus to apply the change in */etc/sevone/data-bus/application.conf* file.

Please refer application.conf example for details on **ssl.truststore**.<options>.

50.1.2.5.2 SSL + SASL Authentication

SASL authentication requires the following.

- Change **security.protocol** setting to **SASL_SSL**.
- Add **`sasl.mechanism`** property to define the SASL method to use.

- Add **sasl.jaas.config** property that SevOne Data Bus can use to authenticate to the broker. Since the value of the property has embedded spaces it must be surrounded by quotes.

  ⚠ Any embedded quotes must be escaped with a **backslash**(\ ).

Please **restart** SevOne Data Bus to apply the change in */etc/sevone/data-bus/application.conf* file.

⚠ Please refer application.conf example for details on **security.protocol**, **sasl.mechanism**, and **sasl.jaas.config**.

50.1.2.5.3 PLAIN Text Authentication

The PLAIN text mechanism simply sends a plaintext username and password. Because the credentials are sent as plaintext, this mechanism must only be used with **SSL**.

Let's assume the username is *sevone* and the password is *sevone_secret*. The following properties will be added or changed in SevOne Data Bus configuration file.

Example

```
...
security.protocol=SASL_SSL
ssl.truststore.location=...
ssl.truststore.password=...

sasl.mechanism=PLAIN

sasl.jaas.config="org.apache.kafka.common.security.plain.PlainLoginModule required \
username=\"sevone\" password=\"sevone_secret\""
...
```

Please **restart** SevOne Data Bus to apply the change in */etc/sevone/data-bus/application.conf* file.

⚠ Please refer application.conf example for details on **security.protocol**, **ssl.truststore**.<options>, **sasl.mechanism**, and **sasl.jaas.config**.

50.1.2.5.4 GSSAPI Authentication

GSSAPI uses **Kerberos** to authenticate. A Kerberos keytab file must be present on SevOne NMS system for this mechanism to work.

Let's assume the keytab is at **/etc/sevone/data-bus/mykeytab** and the principal to use is myuser@EXAMPLE.COM. The properties must look as the following.

Example

```
...
security.protocol=SASL_SSL
ssl.truststore.location=...
ssl.truststore.password=...

sasl.mechanism=GSSAPI

sasl.jaas.config="com.sun.security.auth.module.Krb5LoginModule required \
useKeyTab=true storeKey=true keyTab=\"/etc/sevone/data-bus/mykeytab.keytab\" \
```

```
principal=\"myuser@EXAMPLE.COM\""
...
```

Please **restart** SevOne Data Bus to apply the change in */etc/sevone/data-bus/application.conf* file.

> ⚠ Please refer application.conf example for details on **security.protocol**, **ssl.truststore**.<options>, **sasl.mechanism**, and **sasl.jaas.config**.

50.1.2.5.5 SCRAM Authentication

Kafka can use the SCRAM mechanism with either **SHA-256** or **SHA-512**.

Let's assume the mechanism is *SCRAM-SHA-256*, the username is *sevone*, and the password is *sevone_secret*. The properties can be set as shown below.

Example

```
...
security.protocol=SASL_SSL
ssl.truststore.location=...
ssl.truststore.password=...

sasl.mechanism=SCRAM-SHA-256

sasl.jaas.config="org.apache.kafka.common.security.scram.ScramLoginModule required \
username=\"sevone\" password=\"sevone_secret\""
...
```

Please **restart** SevOne Data Bus to apply the change in */etc/sevone/data-bus/application.conf* file.

> ⚠ Please refer application.conf example for details on **security.protocol**, **ssl.truststore**.<options>, **sasl.mechanism**, and **sasl.jaas.config**.

50.1.2.5.6 OAUTHBEARER Authentication

While both Apache Kafka and Confluent Platform implement an OAUTHBEARER mechanism, it is not for use with client authorization.

> ⓘ **Apache Kafka states the following about OAUTHBEARER...**
>
> The default OAUTHBEARER implementation in Kafka creates and validates Unsecured JSON Web Tokens and is only suitable for use in non-production Kafka installations. For details, please refer to https://kafka.apache.org/documentation/#security_sasl_oauthbearer

Confluent Platform essentially states the same.

> **IMPORTANT**
>
> Do not use token services or the OAUTHBEARER SASL mechanism, listener.name.*rbac.sasl.enabled.mechanisms=OAUTHBEARER*, for external client communications.
> With Role-Based Access Control (RBAC) enabled, token services are intended for internal communication between Confluent Platform components only. For example, it is valid for a Schema Registry licensed client, and not for long-running service principals or client authentication.
> The OAUTHBEARER setting is for internal-use and subject to change, and does not implement a full-featured OAuth protocol.

50.1.3 Stop SevOne Data Bus

To stop SevOne Data Bus, execute one of the following commands:

Gentoo SevOne NMS:

```
$ /etc/init.d/sevone-data-bus stop
```

CentOS SevOne NMS:

```
$ /usr/bin/supervisorctl stop SevOne-data-bus
```

50.2 Common Administrative Tasks

50.2.1 View Logs

You can find SevOne Data Bus logs in **/var/log/SevOne-data-bus**. The current log is **data-bus.log**. Previous logs are rolled up as **data-bus.{yyyy-mm-dd}.{#}.log** (for example, data-bus.2017-10-14.1.log), where **#** is the number of rolled up logs for the specified date.

50.2.2 Configure Logs

Perform the following steps to configure log files.

1. Using the text editor of your choice, open **logback.xml** in /etc/sevone/data-bus.
2. To change the file name format, edit the following line. You can replace **data-bus** with a new name and change the current date format, **yyyy-MM-dd**, to a different date format.

```
<fileNamePattern>
  ${logDir}/data-bus.%d{yyyy-MM-dd}.%i.log
</fileNamePattern>
```

3. To change the log rollup, edit one or more of the following settings. By default, logs are rolled up at least once per day–more often if files exceed the maximum file size.
   - Maximum file size. Replace **{file size}** with the desired file size.

     ```
     <maxFileSize>{file size}</maxFileSize>
     ```

   - Maximum number of logs to roll up. Replace **{number}** with the desired number of logs.

     ```
     <maxHistory>{number}</maxHistory>
     ```

   - Total size of the rollup file. Replace **{total size}** with the desired size.

     ```
     <totalSizeCap>{total size}</totalSizeCap>
     ```

4. To change the log level, edit the following line. Replace **{level}** with **TRACE**, **DEBUG**, **INFO**, **WARN**, or **ERROR**.

```
<root level="{level}">
```

5. Save and close the file.

50.2.3 Check the SevOne Data Bus Status Page

Execute the following command to check the system throughput using the SevOne Data Bus status page. You will need to provide the hostname or IP addres of the SevOne NMS appliance or cluster as well as the port number that the SevOne Data Bus status page runs on. The status page is only available when it is enabled and the port it runs on is defined. You can configure these settings in the **status** entry of the **application.conf** file (see Configure and Start SevOne Data Bus). Details for all metrics can be found in https://kafka.apache.org/documentation.html.

```
$ wget {SevOne NMS hostname or IP address}:{port number}/status
```

For example:

```
$ wget 10.128.18.52:8082/status
```

50.3 SevOne Data Bus and SevOne Data Cloud

To establish data transfer from SevOne NMS to SevOne Data Cloud using SDB as an interface, execute the following steps.

50.3.1 Obtain API Token

1. Go to the SevOne Data Cloud page at http://<SevOne Data Cloud>/api/key/. Replace **<SevOne Data Cloud>** with the hostname or IP address of your SevOne Data Cloud.
2. Save the **API key** it returns.

Example: https://cloud.sevone.com/api/key

```
P5ALkFQ4UA4MTDYu02oODzQ4o2De02_2PaPYXN6bbLs
```

50.3.2 Enable and Configure Internal Datad/Kafka NMS Subscription

To **enable** and **configure** the internal Datad/Kafka NMS Subscription between SevOne Data Bus and SevOne Data Cloud, the following configuration must be added in **/etc/sevone/data-bus/application.conf**.

Example

Datad/Kafka Connection

```
# Configure the datad/kafka connection information
  nms {
    kafka {
      url = "127.0.0.1:9092"
      group = "sdb_group"
      serverProperties = "/etc/kafka/kafka-server.properties"
    }
  }
```

This configuration should not be changed unless if you are running the internal Kafka on a **different** port. In that case, you must modify only the **url** config entry in this configuration.

Example: Configure publishing to Data Cloud

```
# Output configuration
output {
    # Default settings for publishers, which can be overwritten by each publisher
    default {
        type = "dataEngine"
        # When publish errors occur log only every Nth consecutive exception
        # Default to be 50000
        logNthException: 50000

        # Customerize the message key format if needed. Available fields include:
        #     deviceId, deviceName, deviceIp, peerId, objectId, objectName,
        #     objectDesc, pluginId, pluginName, indicatorId, indicatorName,
        #     format, value, time, clusterName, peerIp
        # Default format is "deviceId:objectId".
        key-fields = ["deviceId", "objectId"]
        key-delimiter = ":"
    }


    publishers = [
      {
        # Publisher name will be used to distinguish each publisher in the status page
        # SDB will assign a name for unnamed publishers in the format "publisher-X",
        # where X is the numerical index of unnamed publishers.
        name = "dataEngine"
        type = "dataEngine"
        project = "<your SDB project name - for example, SDB-Cloud>"

        transport = "de-grpc"
        logLevel = "DEBUG"
        grpcHost = "datacloud.turbonomic.io"
        grpcPort = "443"
        grpcToken = "<the API key for the user's account on datacloud.turbonomic.io>>"
        grpcTokenType = "apikey"
        grpcTls = true
        grpcInsecureTls = true
        grpcCertPath = ""
        grpcTlsServerName = ""
        grpcClientCertPath = ""
        grpcClientKeyPath = ""
        collectorName = "<a user-defined collector name>"
        projectId = "<a project name for the user's account on datacloud.turbonomic.io>>"

        # Multiple filters are combined with logical AND.
        filters = [
            "default"
        ]

        topic = "sdb"

        # Kafka producer configuration options.
        # See https://kafka.apache.org/documentation, section 3.3 Producer Configs
        producer {
            # If bootstrap.servers is not defined, SDB will look for the bootstrap.servers
            # defined in output.default.producer.
            #bootstrap.servers = "kafka:9092"
        }
      }
     ]
    }
```

project is for SDB project.
project-id is for Data Cloud project. Value must be in lower-case only and with no spaces. Hyphens are allowed.
Assign the value Obtain API Token step returns to **grpcToken** as shown in the example above.

50.3.3 Export to SevOne Data Cloud

grpcHost

Please refrain from your https:// before the hostname on the **grpcHost** setting.

SevOne Data Bus supports exporting data to SevOne Data Cloud.

Example: application.conf for exporting data to SevOne Data Cloud

Connection configuration can be found under the **publishers** section.

```
# Configure the REST API connection information
api {
    url = "http://127.0.0.1:8080"
        # REST API timeouts in milliseconds
        connectionTimeoutMS = 5000
        # Default to 30 minutes
        socketTimeoutMS = 1800000

    authentication {
        username = "admin"
        password = "SevOne"
        refreshPeriodSeconds = 600
    }

    cache {
        initialLoad = true
        refreshPeriodSeconds = 1800
        expirationMinutes = 180

        pageSizes {
            devicesWObjectsAndIndicators = 10
            indicatorTypes = 100
            objectGroups = 100
            objects = 100
            plugins = 50
        }
    }
}

# Configure the datad connection information
nms {
    kafka {
        url = "127.0.0.1:9092"
        group = "sdb_group2"
        serverProperties = "/etc/kafka/kafka-server.properties"
    }
}

# Configure the Data Bus output format
databus {
```

```
    output {
        format = "avro"
    }
    clusterName = "NMS"
    schemaFile = "/etc/sevone/data-bus/databusmsg.json"
}

# Configure the settings for Schema Registry server if needed
schema-registry {
    #url: "SCHEMA_REGISTRY_SERVER_HOST:SCHEMA_REGISTRY_SERVER_PORT"

    # Subject for the schemas
    subject: "sevone-sdb"
}

# Filters for the Data Bus output
# Note: SDB only exports data from the local peer, so these filters only apply to local data.
filters = [
    {
        name = "default"
        # Specify your filters as different elements in this array
        # by specifying an ID that you would like to be included.
        # A value of -1 is interpreted as any ID.

        # Each column in a given filter is combined with logical AND.
        # Multiple filters are combined with logical OR.
        rules = [
            # Example: Include everything
            {devGrpID = -1, objGrpID = -1, devID = -1, objID = -1}

            # Example: Include only devices 5 and 6
            # {devGrpID = -1, objGrpID = -1, devID = 5, objID = -1},
            # {devGrpID = -1, objGrpID = -1, devID = 6, objID = -1}

            # Example: Include device 5 only if it's in device group 2 and device 6
            # {devGrpID = 2, objGrpID = -1, devID = 5, objID = -1},
            # {devGrpID = -1, objGrpID = -1, devID = 6, objID = -1}

            # Example: Include only objects 2, 3, and 4 from device 5
            # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 2},
            # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 3},
            # {devGrpID = -1, objGrpID = -1, devID = 5, objID = 4}
        ]
    }
]

# Output configuration
output {
    # Default settings for publishers, which can be overwritten by each publisher
    default {
        type = "dataEngine"
        # When publish errors occur log only every Nth consecutive exception
        # Default to be 50000
        logNthException: 50000

        # Customerize the message key format if needed. Available fields include:
        #     deviceId, deviceName, deviceIp, peerId, objectId, objectName,
        #     objectDesc, pluginId, pluginName, indicatorId, indicatorName,
        #     format, value, time, clusterName, peerIp
        # Default format is "deviceId:objectId".
        key-fields = ["deviceId", "objectId"]
        key-delimiter = ":"

        # Kafka producer configuration options.
        # See https://kafka.apache.org/documentation, section 3.3 Producer Configs
        producer {
            acks = "-1"
            retries = "0"
```

```
            linger.ms = "10"
            batch.size = "1000000"
            request.timeout.ms = "60000"
            max.in.flight.requests.per.connection = "2"
        }
    }

    publishers = [
        {
            # Publisher name will be used to distinguish each publisher in the status page
            # SDB will assign a name for unnamed publishers in the format "publisher-X",
            # where X is the numerical index of unnamed publishers.
            name = "dataEngine"
            type = "dataEngine"
            #name = "debug"
            #type = "debug"
            project = "SDB-Cloud"

            transport = "de-grpc"
            logLevel = "DEBUG"
            # URL of destination cloud system
            grpcHost = "datacloud.turbonomic.io”
            grpcPort = "443"
            # Get this from your Data Cloud Service Profile
            grpcToken = "VT7bai1Lswut1xZN62imihUBirW7fI9MFOou0CfMfkz"
            grpcTokenType = "apikey"
            grpcTls = true
            grpcInsecureTls = false
            grpcCertPath = ""
            grpcTlsServerName = ""
            grpcClientCertPath = ""
            grpcClientKeyPath = ""
            # Streams all data under this tag in data cloud (this tag can be customer
defined)
            collectorName = "SDB"
            # Your project ID Assigned by SevOne (get this from your Data Cloud Service
Profile)
            projectId = "sevone"

            # Multiple filters are combined with logical AND.
            filters = [
                "default"
            ]

            topic = "sdb"

            # Kafka producer configuration options.
            # See https://kafka.apache.org/documentation, section 3.3 Producer Configs
            producer {
                # If bootstrap.servers is not defined, SDB will look for the
bootstrap.servers
                # defined in output.default.producer.
                #bootstrap.servers = "kafka:9092"
            }
        }
    ]
}

status {
        # Write SDB status metrics to log every 300 seconds
        metricsLogInterval: 300
    http {
        # Configure the status HTTP page created by the Data Bus
        # hosted at http://hostname:port/status
        enabled: true
        port: 8082
    }
    https {
        # Configure the status HTTPS page created by the Data Bus
```

```
        # hosted at https://hostname:secure_port/status
        enabled: false
        secure_port: 8443
        # Use the java keytool utility to create the keystore, generate a certificate or
import an existing certificate.
        # keytool -genkey -alias alias -keypass "your private key password here"
-keystore /etc/sevone/data-bus/sdb.keystore -storepass "your kseystore password here"
        private_key_password: "password"
        keystore_password: "password"
        keystore_path: "/etc/sevone/data-bus/sdb.keystore"
    }
}

# The Data Bus's threading is done with the Akka framework.
# Below you can granularly adjust the framework's behavior.
# You can adjust the types of pools and the dispatcher that we use,
# as well as control the parallelization factors.
# See Akka's 2.5 documentation for more in-depth information.
akka {
    # This is to use SLF4J logger that will log actor logs to file.
    loggers = ["akka.event.slf4j.Slf4jLogger"]
    loglevel = "DEBUG"

    actor {
        deployment {
            # This is the set of actors that parse datad protobufs to Scala objects
            "/*/decodeActorRouter" {
                router = balancing-pool
                nr-of-instances = 10
            }

            # This is the set of actors that encode Scala objects to JSON or Avro
            "/*/encodeActorRouter" {
                router = balancing-pool
                nr-of-instances = 10
            }

            # This is the set of actors the add metadata to indicator data
            "/*/enrichActorRouter" {
                router = balancing-pool
                nr-of-instances = 10
            }

            # This is the set of actors that send the enriched objects to Kafka
            "/*/publishActorRouter" {
                router = balancing-pool
                nr-of-instances = 30
            }
        }

        # This is the Akka dispatcher that we use for the above pools
        data-bus-dispatcher {
            # Dispatcher is the name of the event-based dispatcher
            type = Dispatcher
            # What kind of ExecutionService to use
            executor = "affinity-pool-executor"
            # Configuration for the thread pool
            affinity-pool-executor {
                # Min number of threads to cap factor-based parallelism number to
                parallelism-min = 2
                # Parallelism (threads) ... ceil(available processors * factor)
                parallelism-factor = 1
                # Max number of threads to cap factor-based parallelism number to
                parallelism-max = 4
            }
            # Throughput defines the maximum number of messages to be
            # processed per actor before the thread jumps to the next actor.
            # Set to 1 for as fair as possible.
            throughput = 1
```

```
242             }
243         }
244
245         scheduler{
246             #tick-duration = 1ms
247         }
248     }
```

The URL for the destination cloud system must be set to **datacloud.turbonomic.io**.

Example

```
grpcHost = "datacloud.turbonomic.io"
```

50.3.3.1 Proxy Configuration

A web proxy may be used to allow the internal devices of the user to have access to the external internet. For example, allow on-premises SevOne NMS + SevOne Data Bus to export data to a Data Cloud instance.

To configure and use the proxy, execute the following steps.

1. Using a text editor of your choice, edit **/etc/supervisord.d/SevOne-data-bus.ini** file.

```
$ vi /etc/supervisord.d/SevOne-data-bus.ini
```

2. Set the **environment** variable and save **/etc/supervisord.d/SevOne-data-bus.ini** file.

```
environment=HTTP_PROXY=http://<host name or IP address>:<port number>
```

Example

```
environment=HTTP_PROXY=http://proxy.sevone.com:8080

OR

environment=HTTP_PROXY=http://10.128.10.11:8080
```

3. Execute the following command to update.

```
$ /usr/bin/supervisorctl update SevOne-data-bus
```

50.4 Enable OpenTracing

SevOne Data Bus supports OpenTracing. Using a text editor of your choice, ddd the following to **/etc/sevone/data-bus/application.conf** file. All Jaeger environment options are supported.

```
opentracing {
  JAEGER_DISABLED = "false"
  JAEGER_SAMPLER_TYPE = "const"
  JAEGER_SAMPLER_PARAM = "1"
  JAEGER_SERVICE_NAME = "SDB"
  JAEGER_PROPAGATION = "b3"
  JAEGER_AGENT_HOST = "localhost"
  JAEGER_AGENT_PORT = "6831"
}
```

Ensure local Jaeger instance is running

```
$ supervisorctl start jaeger
```

50.5 SevOne Data Bus Historical Backfill

50.5.1 Overview

This feature will enable you to republish historical data from a specified time period through sending a POST request using the REST API endpoint. Both start time and end time is formatted in Unix timestamp in seconds. This feature is introduced in SevOne Data Bus 1.4 version, in which modifications to the configuration file are required.

50.5.1.1 Use Cases

The ability to backfill data in case something goes wrong (communication issues with Kafka cluster, etc.) and have that data sent across the same Kafka setup as you do for polled data.

50.5.2 Configuration

To enable backfill data, an external broker must be set up to receive historical data and SevOne Data Bus must be able to distinguish this broker from others. The logic of the implementation is to add a flag into the historical broker setting to enable SevOne Data Bus to recognize it. The modifications of configurations are listed below.

Republish actor is refactored so that it will not overload the system while republishing. Configuration of **backfillSleepTime** is used to republish data. This allows the republish actor to take a rest between republishing of each indicator.

50.5.2.1 Application.conf

Application.conf(v1.3)

```
1   api = ${api}
2
3   nms = ${nms}
4
5   # Configure the Data Bus output format
6   databus = ${databus}
7
8   # Configure the settings for Schema Registry server if needed
9   schema-registry = ${schema-registry}
10
```

```
    # Filters for the Data Bus output
    # Note: SDB only exports data from the local peer, so these filters only apply to local data.
    filters = ${filters}

    # Output configuration
    output = ${output}

    status = ${status}

    akka = ${akka}
```

Application.conf(current)

```
DATABUS {
  api = ${api}

  nms = ${nms}

  # Configure the Data Bus output format
  databus = ${databus}

  # Configure the settings for Schema Registry server if needed
  schema-registry = ${schema-registry}

  # Filters for the Data Bus output
  # Note: SDB only exports data from the local peer, so these filters only apply to local data.
  filters = ${filters}

  # Output configuration
  output = ${output}

  status = ${status}
}

akka = ${akka}
```

Backfill Throttling

```
# Historical data is republished indicator based, this setting controls how long republish actor will
sleep before republishing next indicator
# It should be added inside databus section inside DATABUS block in application.conf for version after 1.4
# Added to databus section in conf file for version before 1.4
DATABUS {
        ......
    databus {
        ......
        backfillSleepTime = 0 # millisecond
        .......
    }
       ......
}
```

50.5.2.2 Kafka Configurations

For historical data broker, you need to add an **isLive** flag and set the value to **false** to let SevOne Data Bus recognize this historical broker. The rest of the configuration is the same as others.

Application.conf(current)

```
# Output configuration
  output {
    # Default settings for publishers, which can be overwritten by each publisher
    default {
      # Publisher types:
      #   kafka, pulsar, dataEngine
      # Default to be kafka
      type = "kafka"

      # When publish errors occur log only every Nth consecutive exception
      # Default to be 50000
      logNthException: 50000

      # Kafka section

      # Customerize the message key format if needed. Available fields include:
      #     deviceId, deviceName, deviceIp, peerId, objectId, objectName,
      #     objectDesc, pluginId, pluginName, indicatorId, indicatorName,
      #     format, value, time, clusterName, peerIp
      # Default format is "deviceId:objectId".
      key-fields = ["deviceId", "objectId"]
      key-delimiter = ":"

      # Kafka producer configuration options.
      # See https://kafka.apache.org/documentation, section 3.3 Producer Configs
      kafka-producer {
        acks = "-1"
        retries = "0"
        linger.ms = "10"
        batch.size = "1000000"
        request.timeout.ms = "60000"
        max.in.flight.requests.per.connection = "2"
      }
    }

    publishers = [
      {
        name = "default-producer"
        filters = [
          "default"
        ]
        topic = "<Enter your topic name. For example, sdb>"

        producer {
          bootstrap.servers = "localhost:9092"
          security.protocol=SSL
          ssl.truststore.location="<path to trust store. For example, /var/private/ssl/kafka.client.truststore.jks>"
          ssl.truststore.password="<trust store password. For example, abc123>"
          ssl.protocol="<The SSL protocol allowed in JVMs. For example, TLSv1.2, TLSv1.3>"
          ssl.endpoint.identification.algorithm=""

          # for SSL, uncomment the lines below
          # ssl.keystore.location="<path to key store. For example, /var/private/ssl/kafka.client.keystore.jks>"
          # ssl.keystore.password="<key store password. For example, test1234>"
          # ssl.key.password="<key password. For example, test1234>"

          # for Kerberos, uncomment the line below
          # sasl.jaas.config: "com.sun.security.auth.module.Krb5LoginModule required useKeyTab=true storeKey=true keyTab=\"/path/to/kafka/client/keytab\" principal=\"kafka-client-1/{hostname}@TEST.COM\";"

          # for SCRAM, uncomment the lines below
          # sasl.mechanism=SCRAM-SHA-256
```

```
          # security.protocol=SASL_PLAINTEXT
          # sasl.jaas.config="""org.apache.kafka.common.security.scram.ScramLoginModule
required username="sdbuser" password="sdb-secret";"""
        }
      },
      {
        name = "default-producer-historical"
        filters = [
          "default"
        ]
        topic = "<Enter your topic name. For example, sdb-historical>"
       isLive = false

        producer {
          bootstrap.servers = "localhost:9092"
        }
      }
    ]
  }
```

50.5.2.3 Pulsar Configurations

```
# Output configuration
output {
    # Default settings for publishers, which can be overwritten by each publisher
    default {
       # Publisher types:
       #   kafka, pulsar, dataEngine
        # Default to be pulsar
        type = "pulsar"

        # When publish errors occur log only every Nth consecutive exception
        # Default to be 50000
        logNthException: 50000

        # Pulsar section

        # Customerize the message key format if needed. Available fields include:
        #   deviceId, deviceName, deviceIp, peerId, objectId, objectName,
        #      objectDesc, pluginId, pluginName, indicatorId, indicatorName,
        #      format, value, time, clusterName, peerIp
        # Default format is "deviceId:objectId".
        key-fields = ["deviceId", "objectId"]
        key-delimiter = ":"

        # Pulsar producer configuration options.
        pulsar-producer {
            batchingMaxMessages = 100000 # millisecond
            sendTimeoutMs = 30000 #millisecond
        }
     }

     publishers = [
        {
            # Publisher name will be used to distinguish each publisher in the status page
            # SDB will assign a name for unnamed publishers in the format "publisher-X",
            # where X is the numerical index of unnamed publishers.
            name = "pulsar-producer"

            # This field is important
            type = "pulsar"

            # Multiple filters are combined with logical AND.
            filters = [
                "default"
            ]
```

```
45
46              tenant = "public"
47              namespace = "default"
48              topic = "<Enter your topic name. Eor example, sdb-pulsar>"
49              topic-type = "persistent"
50
51              # Pulsar producer configuration options.
52              # See https://pulsar.apache.org/docs/en/2.4.2/client-libraries-java/
53              client {
54                  # If bootstrap.servers is not defined, SDB will look for the bootstrap.servers
55                  # defined in output.default.producer.
56                  serviceUrl = "pulsar+ssl://<Enter IP address>:<Enter port number>/" OR <can be
    set to null>
57                  useTls = true
58                  tlsTrustCertsFilePath = "<Enter your path to>/ca.cert.pem"
59                  tlsAllowInsecureConnection = true
60              }
61
62              producer {
63                  # Check http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/
    ProducerBuilder.html#compressionType-org.apache.pulsar.client.api.CompressionType-
64                  # for available conf
65                  compressionType = "ZLIB"
66                  batchingEnabled = true
67                  messageRoutingMode = "SinglePartition"
68              }
69          },
70          {
71              name = "pulsar-producer-historical"
72
73              # This field is important
74              type = "pulsar"
75
76              # Multiple filters are combined with logical AND.
77              filters = [
78                  "default"
79              ]
80
81              isLive = false
82
83              tenant = "public"
84              namespace = "default"
85              topic = "<Enter your topic name. For example, sdb-pulsar-historical>"
86              topic-type = "persistent"
87
88
89              # Pulsar producer configuration options.
90              # See https://pulsar.apache.org/docs/en/2.4.2/client-libraries-java/
91              client {
92                  # If bootstrap.servers is not defined, SDB will look for the bootstrap.servers
93                  # defined in output.default.producer.
94                  serviceUrl = "pulsar+ssl://<Enter IP address>:<Enter port number>/" OR <can be
    set to null>
95                  useTls = true
96                  tlsTrustCertsFilePath = "<Enter your path to>/ca.cert.pem"
97                  tlsAllowInsecureConnection = true
98              }
99
100             producer {
101                 # Check http://pulsar.apache.org/api/client/2.4.2/org/apache/pulsar/client/api/
    ProducerBuilder.html#compressionType-org.apache.pulsar.client.api.CompressionType-
102                 # for available conf
103                 compressionType = "ZLIB"
104                 batchingEnabled = true
105                 messageRoutingMode = "SinglePartition"
106               }
107           }
108     ]
109 }
```

50.5.2.4 DataEngine Configurations

```
# Output configuration
output {
    # Default settings for publishers, which can be overwritten by each publisher
    default {
        # Publisher types:
        #   kafka, pulsar, dataEngine
        # Default to be dataEngine
        type = "dataEngine"

        # When publish errors occur log only every Nth consecutive exception
        # Default to be 50000
        logNthException: 50000

        # Customerize the message key format if needed. Available fields include:
        #     deviceId, deviceName, deviceIp, peerId, objectId, objectName,
        #     objectDesc, pluginId, pluginName, indicatorId, indicatorName,
        #     format, value, time, clusterName, peerIp
        # Default format is "deviceId:objectId".
        key-fields = ["deviceId", "objectId"]
        key-delimiter = ":"
    }

    publishers = [
        {
            # Publisher name will be used to distinguish each publisher in the status page
            # SDB will assign a name for unnamed publishers in the format "publisher-X",
            # where X is the numerical index of unnamed publishers.
            name = "dataEngine-producer"


            # This field is important
            type = "dataEngine"

            project = "SDB-Cloud"
            transport = "de-grpc"
            logLevel = "DEBUG"


            # URL of destination cloud system
            grpcHost = "datacloud.turbonomic.io"
            grpcPort = "443"

            # Get this from your Data Cloud Service Profile
            grpcToken = "VT7bai1Lswut1xZN62imihUBirW7fI9MFOou0CfMfkz"
            grpcTokenType = "apikey"
            grpcTls = true
            grpcInsecureTls = false
            grpcCertPath = ""
            grpcTlsServerName = ""
            grpcClientCertPath = ""
            grpcClientKeyPath = ""

            # Streams all data under this tag in data cloud (this tag can be customer defined)
            collectorName = "<Enter your topic name. For example, sdb-dataengine>"

            # Your project ID Assigned by SevOne (get this from your Data Cloud Service
Profile)
            projectId = "sevone"

            # Multiple filters are combined with logical AND.
            filters = [
                "default"
            ]

            topic = "sdb"

```

```
            # DataEngine producer configuration options.
            producer {}
        },
        {
            # Publisher name will be used to distinguish each publisher in the status page
            # SDB will assign a name for unnamed publishers in the format "publisher-X",
            # where X is the numerical index of unnamed publishers.
            name = "dataEngine-producer-historical"


            # This field is important
            type = "dataEngine"

            project = "SDB-Cloud"
            transport = "de-grpc"
            logLevel = "DEBUG"


            # URL of destination cloud system
            grpcHost = "datacloud.turbonomic.io"
            grpcPort = "443"

            isLive = false

            # Get this from your Data Cloud Service Profile
            grpcToken = "VT7bai1Lswut1xZN62imihUBirW7fI9MFOou0CfMfkz"
            grpcTokenType = "apikey"
            grpcTls = true
            grpcInsecureTls = false
            grpcCertPath = ""
            grpcTlsServerName = ""
            grpcClientCertPath = ""
            grpcClientKeyPath = ""

            # Streams all data under this tag in data cloud (this tag can be customer defined)
            collectorName = "<Enter your topic name. For example, sdb-dataengine-historical>"


            # Your project ID Assigned by SevOne (get this from your Data Cloud Service
Profile)
            projectId = "sevone"

            # Multiple filters are combined with logical AND.
            filters = [
                "default"
            ]

            topic = "sdb"

            # DataEngine producer configuration options.
            producer {}
        }
    ]
}
```

50.5.3 Usage

HTTP Method	End Point	Parameter	Purpose
POST	http://localhost:PORT/status/republish	**PORT**: port number of http server **Start time**: Unix timestamp in seconds **End time**: Unix timestamp in seconds	Republish data for a given time period

The usage of this feature can be accomplished through the bash script **RepublishHistoricalData.sh** provided in SevOne Data Bus. You will only need to run the script along with start time and end time as two independent parameters. For example:

Bash Command

```
./RepublishHistoricalData.sh 1519000000 1520000000
```

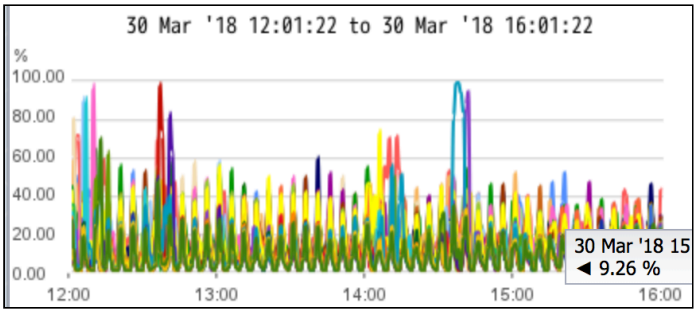
You are also able to use third-party tools such as **Postman** to send the request to the **http** server of SevOne Data Bus.

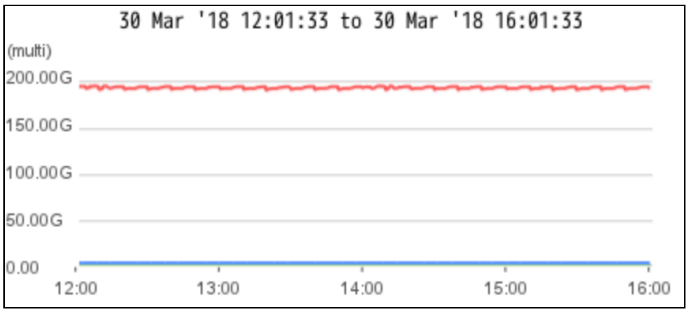
50.5.4 Performance

Example: A CentOS box used for testing with the following environment setup.

CPU	# of CPU	Memory	Swap Memory	Devices	Objects	Indicators
Intel E5-2680@ 2.70GHz	32	252GB	15.6GB	1001	200101	4000578

Due to the restriction of the REST API(Approx 20-50ms response time for each query), the maximum throughput for retrieving indicator data is 50 indicators per second. The system load is monitored while doing republishing, CPU utilization is around 60-80%, free memory is about 65GB of 252GB.





50.5.5 User Manual

1. Add a Kafka publisher in **/etc/sevone/data-bus/application.conf**, add a field called **isLive** inside this publisher and set it to **false**.

```
1  {
```

```
    name = "default-producer"

    filters = [
      "default"
    ]

    isLive = false

    topic = "historical data"

    producer {
      bootstrap.servers = "localhost:9092"
    }
```

2. Find the script called **RepublishHIstoricalData.sh** provided in the **scripts** folder. Make it executable using the following command in the scripts folder.

```
$ chmod 755 ./RepublishHistoricalData.sh
```

3. Execute the script along with two timestamps - denote the start time and end time. Each timestamp should be a Unix timestamp in seconds, the command looks like below.

Bash Command

```
$ ./RepublishHistoricalData.sh [STARTTIME] [ENDTIME]


Example:
$ ./RepublishHistoricalData.sh 1519000000 1520000000
```

The end time must be a timestamp from at least one hour ago compared to the current time.

4. Currently, the republish actor blocks the current job until it completes. If you want to stop the current job, then execute the following command.

```
$ ./RepublishHistoricalData.sh stop
```

50.6 SevOne Data Bus JMX Plugin

50.6.1 Overview

This feature enables you to retrieve runtime metrics of SevOne Data Bus through JMX indicators in SevOne NMS system. It uses the JMX module in SevOne polld, and all the data polled also gets pushed to external Kafka brokers for later use. This feature is introduced in SevOne Data Bus 1.4 version.

50.6.1.1 Use Cases

Monitor SevOne Data Bus for runtime metrics such as EncodeCount, DecodeCount, EncodeDeficit, AvgEncodeTime, etc. For example, create instant graph of these metrics in SevOne NMS.

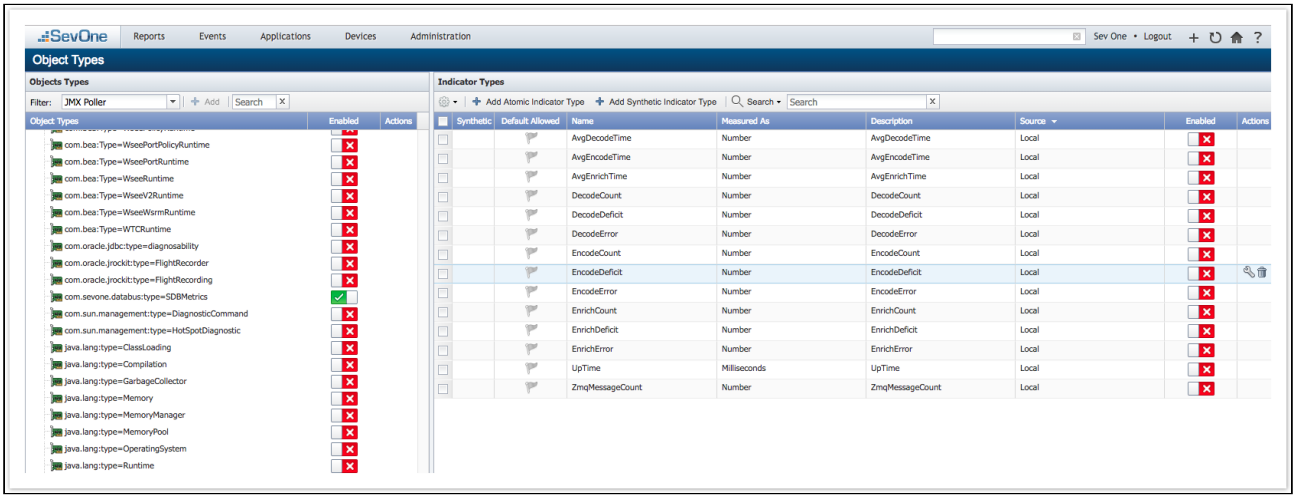
50.6.2 Configuration

To enable JMX plugin for SevOne Data Bus, you need to execute database migration to add SevOne Data Bus JMX plugin information into the database. Moreover, you also need to change SevOne Data Bus config to set up JMX server inside SevOne Data Bus.

50.6.2.1 Database

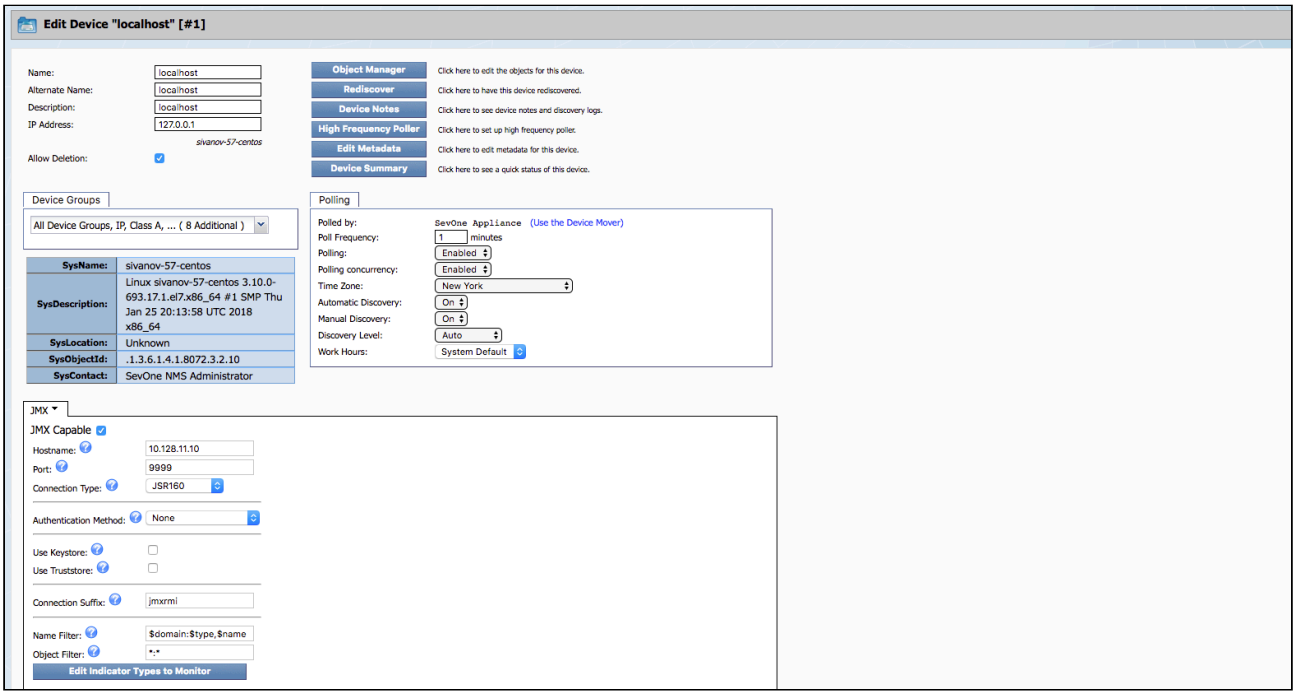
Database migration will be executed by the NMS installation. After successfully executing the migration, the corresponding plugin and indicators will show up in the GUI.

Please go to SevOne NMS GUI > Administration > Monitoring Configuration > Object Types > JMX Poller to enable / disable indicators.



50.6.2.2 Device Manager

Enable JMX poller on the device you want to monitor.



50.6.2.3 application.conf

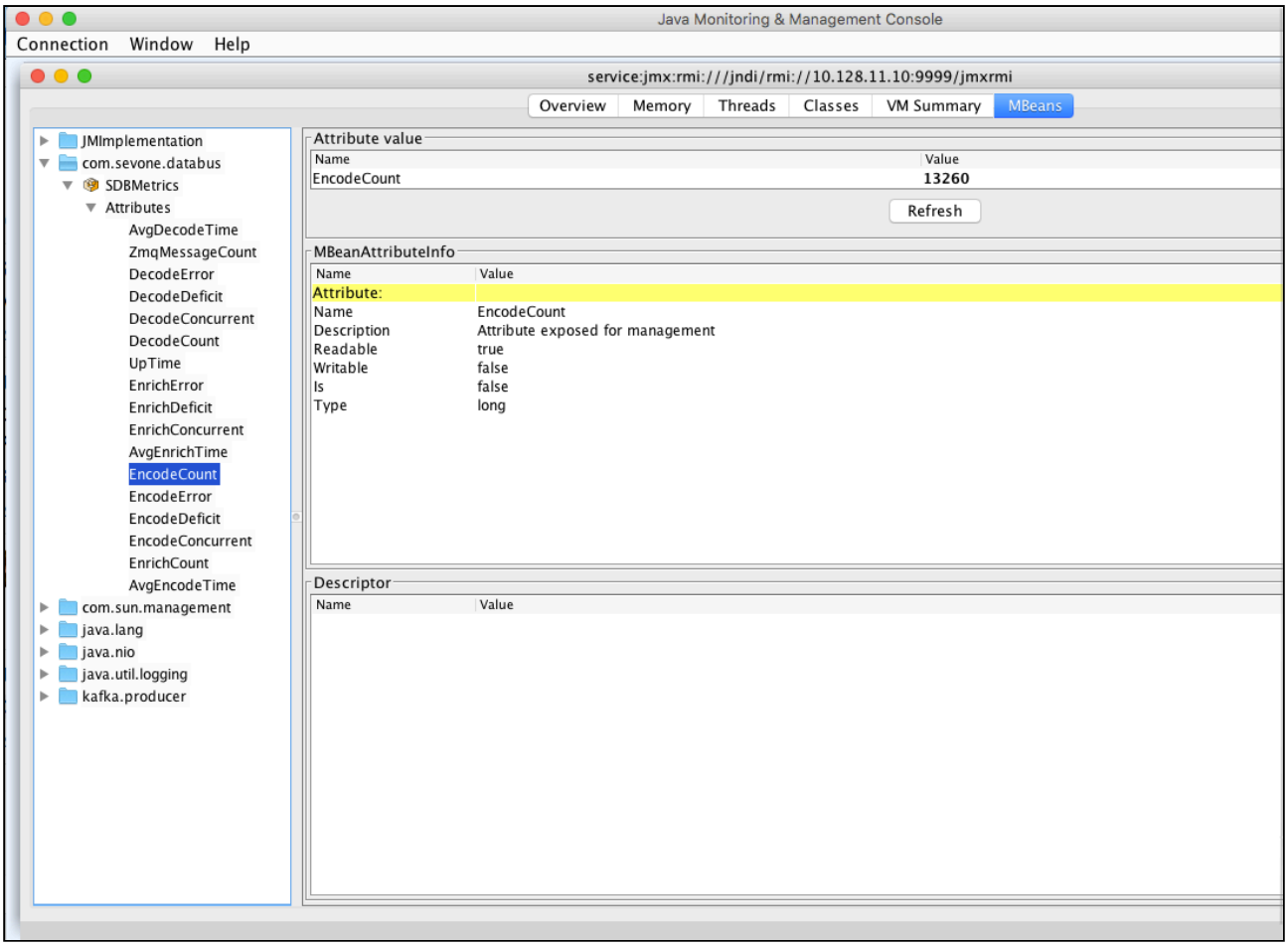
You need to provide SevOne Data Bus with the host IP address and port number used for the JMX server - this will require a new section in config file. An internal server inside SevOne Data Bus will start at the same time SevOne Data Bus starts. This server can also be accessed through other applications such as *JConsole*.

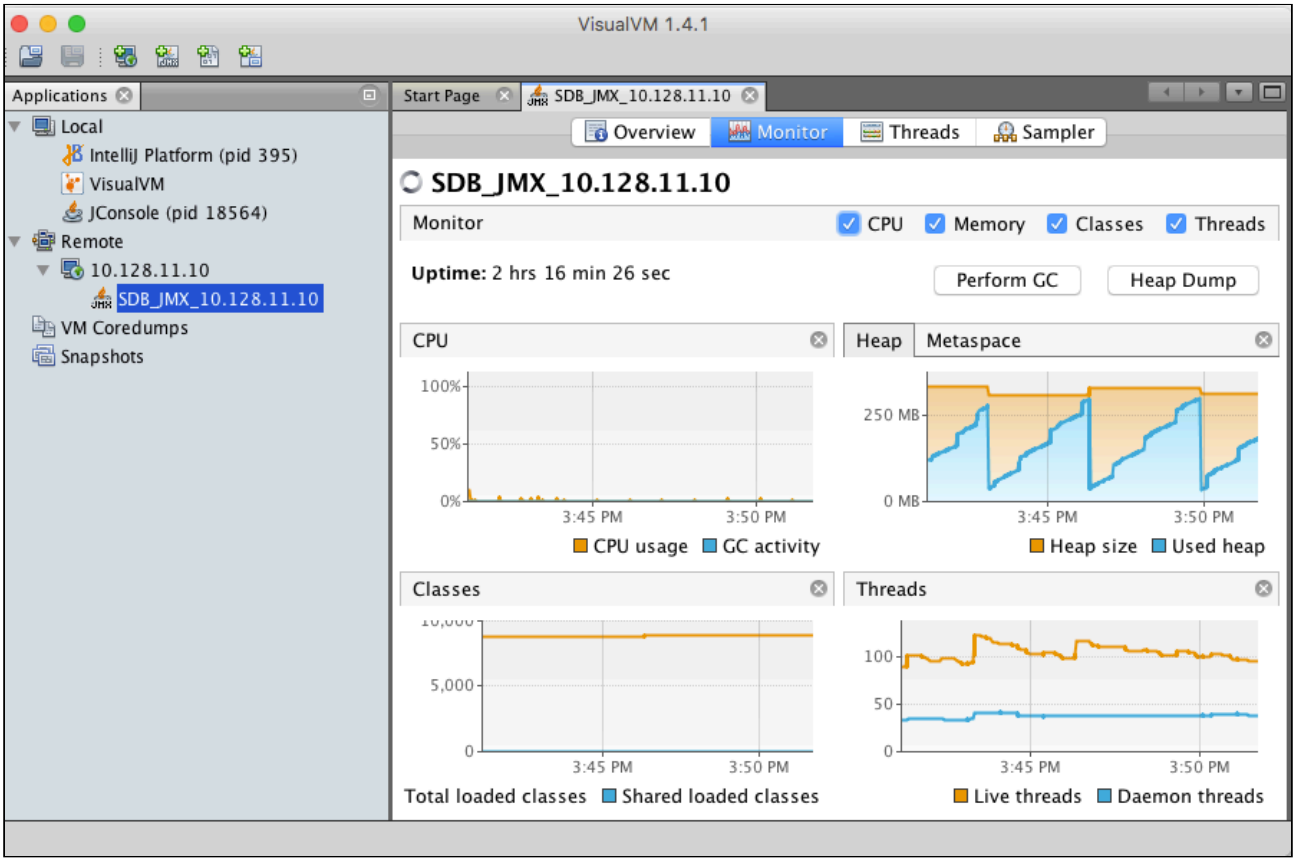
Application.conf(v1.4)

```
version = "1.4"
DATABUS {
    ...
    jmx {
        hostIP = "localhost"
        port = 9999
        enabled = true
    }
    ...
}
    ...
```

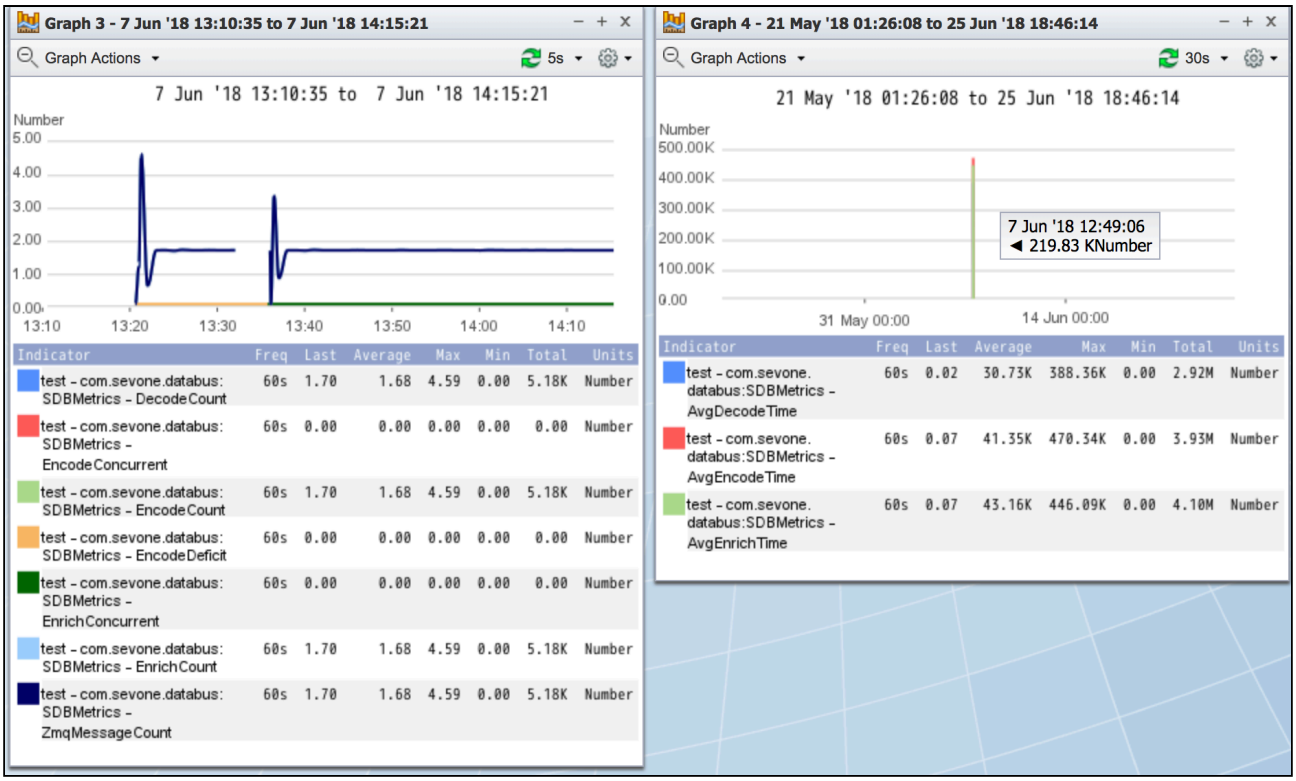
50.6.3 Debug

To make sure the metrics you have retrieved are correct, you can also make use of JConsole or VisualVM to get numbers through JMX server and compare those numbers with what you get from the traditional status page.





50.6.4 GUI View



50.7 SevOne Data Bus Troubleshooting

50.7.1 Configuration File

SevOne Data Bus is controlled by the configuration file located in **/etc/sevone/data-bus/application.conf**. Most common problems with SevOne Data Bus are a result of a bad configuration option. If SevOne Data Bus fails to start, check if:

- the configuration file follows the correct structure from the template file provided
- the comment line incorrectly wraps to start a new line
- there is any **NULL** existing in the config file

50.7.2 Using SDB Status Page

SDB has a built-in web server that can show verbose statistics of its internal operations. The server can be accessed on the local host and the port is configurable in the **/etc/sevone/data-bus/application.conf** file.

50.7.2.1 Verify SevOne Data Bus Status

- Go to SevOne Data Bus status page and check error counts. If a constant increase or a huge number of error counts are observed, a potential issue may be happening or has already happened.
- Check SevOne Data Bus log file.
- use **grep -i 'Error' | grep -v 'SDB Metrics'** to get all error messages
- use **grep 'Exception (count='** for filtering out Kafka publisher related errors

50.7.3 REST API

- If there are any REST API issues, you may restart the REST API with **supervisorctl restart SevOne-restapi**.
- Restart **SevOne-data-bus**
- Ensure the REST API configuration section in the **application.conf** is correct.

50.7.4 Remote Kafka Broker

Manually verify Kafka broker connectivity on the box:

- **telnet BROKER_IP BROKER_PORT**
- Upon successful connection, expect to see message **Connected to BROKER_IP**

50.7.5 Caching

If there are any caching issues, you may restart **SevOne-data-bus**. Upon restart, the application will reload its caches entirely.

50.8 FAQs

50.8.1 How does SevOne Data Bus support HSA?

SevOne Data Bus does not currently run on the HSA. When a failover occurs, it needs to be started manually.

50.8.2 What is the bandwidth needed to stream the data?

52 MB/s (416Mbps) for a fully loaded PAS 200K is needed to stream the data.

50.8.3 What is the maximum supported latency?

Up to 250ms Round-trip Time latency for SevOne Data Bus is tested and supported.

50.8.4 Since SevOne Data Bus delivers all of the data, what scalability testing data is available?

All PAS200k data processing in real-time can be streamed - this is tested and supported.

50.8.5 Does SevOne Data Bus support multiple topics and if so, how many can it support?

SevOne Data Bus supports multiple brokers with a topic. So conceptually, it also supports multiple topics (the same broker configured individually for each topic). There are currently no specific limits tested, provided data is not duplicated across multiple brokers/topics.

50.8.6 Does SevOne Data Bus streaming affect SevOne NMS's ability to poll SNMP data?

The configuration used during scale testing did not impact SevOne NMS's ability to poll data.

50.8.7 SevOne Data Bus connection is lost

50.8.7.1 Will SevOne Data Bus try to reconnect once the connection is lost?

Yes, SevOne Data Bus will try to reconnect once the connection is lost. Kafka producer has **retries** configuration setting. If a connection is lost, Kafka producer retries the number of times **retries** setting is set to.

50.8.7.2 What happens to the messages that should have been streamed during the period when the connection is lost? Are they stored somewhere and retransmitted later on?

SevOne Data Bus's Kafka producer buffers messages for a short time when it is trying to reconnect to the external Kafka. If the connection comes back, the messages will be transmitted correctly, otherwise they are lost.

50.8.7.3 If there is a buffer, how many messages can it hold? What happens when it fills up?

SevOne Data Bus does not have a buffer and it will hold as many messages as possible. The only limitation is how much memory is allocated to JVM for SevOne Data Bus. For Kafka, how many messages can be buffered for each real **send** operation can be controlled. The buffered messages are sent out to Kafka broker when the Kafka send buffer is full.

50.8.8 Can flow data (metrics and/or flows) be exported via Kafka? If so, how can it be enabled?

Flow data (metrics and/or flows) cannot be exported via Kafka. Flows are ingested by DNC whereas metrics are ingested via the PAS. SevOne Data Bus does not see flows at all.

Due to the nature of flows and DNC scale consideration, it is best to redirect the flows to the receiving system because anything on DNC will likely impact the published scale numbers. DNCs are built for scale ingestion and not for publishing.

51 xStats Source Manager

The xStats Source Manager enables you to manage the data collection sources for the xStats plugin. Sources are adapter specific means of collecting xStats data. xStats adapters are manufacturer/equipment specific applications. You can create an unlimited number of xStats sources.

To access the xStats Source Manager from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **xStats Source Manager**.

Data the xStats source collects creates the devices that use the xStats plugin, the xStats object types to discover, and the xStats indicator types to poll on the devices. Additional data about xStats sources appears on the xStats Log Viewer page.

- **Name** - Displays the xStats source name.
- **Peer** - Displays the name of the peer on which the data from the source resides.
- **Retrieval Frequency** - Displays how frequently the source retrieves data.
- **Source Type** - Displays the adapter the source type uses to collect data.
- **Device Creation** - Displays *Automatic* when the devices the source retrieves are automatically added to SevOne NMS and appear on the Device Manager. Displays *Manual* when you can manually add the device to SevOne NMS or you can link the device to a device that already exists in SevOne NMS. A link combines the xStats data from a device the source finds with other data on a device that is already in SevOne NMS.
- **Devices** - Displays the number of devices the source either automatically created or that you can manually add and the number of new devices the xStats source discovered for the source.
- **Unique Files Only** – Displays *Yes* when the source collects only unique files that have yet to be retrieved. Displays *No* when the source collects all files.
- **Last Retrieved** - Displays the date and time the source most recently retrieved data.

51.1 Manage xStats Sources

Perform the following steps to manage xStats sources.

1. Click **Add Source** or click  to display the Add/Edit Source pop-up.
2. In the **Name** field, enter the source name.
3. Click the **Peer** drop-down and select the peer to monitor data from the source.
4. In the **IP Address** field, enter the IP address of the device from which the source is to monitor data.
5. Click the **Source Type** drop-down and select the adapter for the source to use.
6. In the **Retrieval Frequency (seconds)** field, enter the number of seconds for how frequently the source is to attempt to retrieve/receive new data.
7. Click the **Device Creation** drop-down.
    - Select **Manual** to enable manual addition of the devices the source discovers to SevOne NMS and the ability to link the devices to devices that already exist in SevOne NMS.
    - Select **Automatic** to automatically add the devices the source discovers to SevOne NMS on the Device Manager with the xStats plugin enabled.
8. Select the **Unique Files Only** check box to have the source collect only files that have yet to be collected. Clear this check box to collect all available files even if the file was previously collected and to re-process all previous data from those files.
9. In the **Username** field, enter the user name SevOne NMS needs to authenticate onto the device.
10. In the **Password** field, enter the password SevOne NMS needs to authenticate onto the device.
11. In the **Override Retrieval Directory** field, enter the full path to the directory on the SevOne NMS appliance where you prefer to store data. Leave this field blank to accept the default directory.

51.2 Manage xStats Devices

The xStats data from the source creates devices. The Devices column displays the number of devices the source creates and the number of new devices that the latest poll found. When you define a source, you choose to require manual intervention to add the xStats devices to SevOne NMS or to have the source automatically add the devices to SevOne NMS.

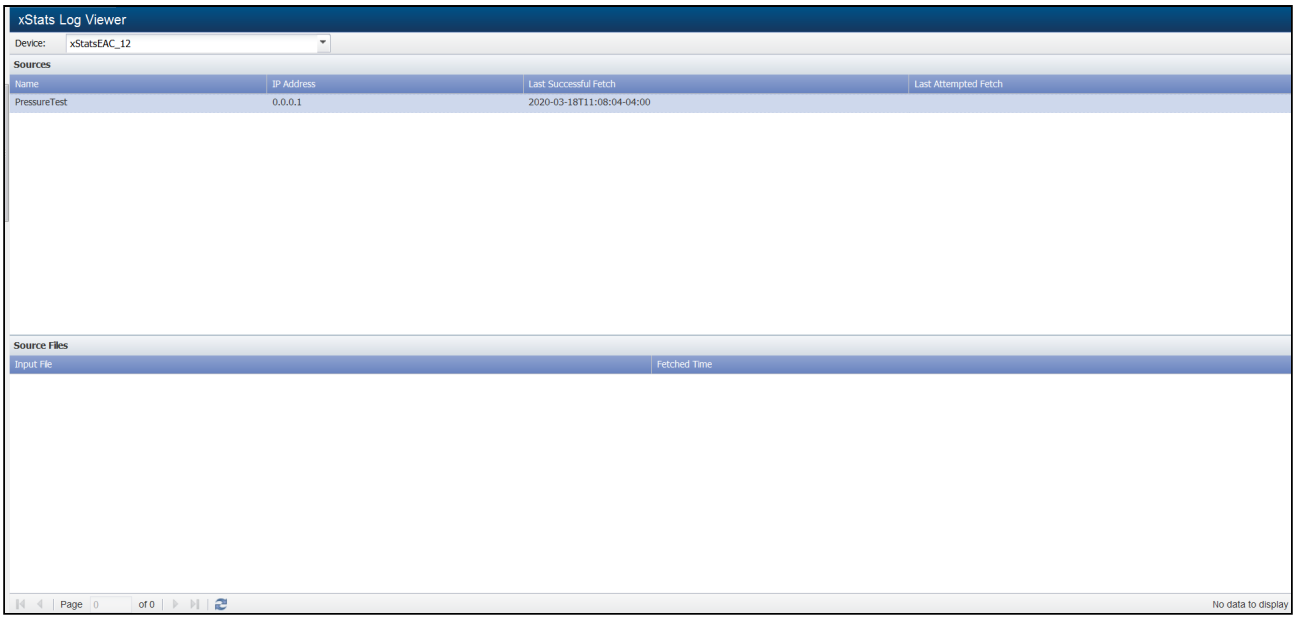
- Click the link in the **Devices** column or click  in the Actions column to display the Manage Devices pop-up.
- Watched displays  when the device is watched by the xStats plugin or displays  when the device is ignored by the xStats plugin.

- Name From Source - Displays the name of the device as discovered by the xStats source.
- IP Address - Displays the IP address of the device.
- First Seen - Displays the date/time the device first appeared from the source.
- Name In SevOne - Displays the name of the device that displays on the Device Manager.
-  - Click to navigate to the Edit Device page where you can edit the device.
-  - Click to display a link to the Device Summary and links to the report templates that are applicable for the device.
-  - When you define the source to require manual device creation, this icon enables you to link the new device to a device that is already in SevOne NMS. This is useful when multiple sources find the same xStats device.
-  - When you define the source to require manual device creation, this icon enables you to add the device as a new device.

52 xStats Log Viewer

The xStats Log Viewer enables you to view additional details from the xStats sources you define on the xStats Source Manager. xStats sources create the xStats devices, objects, and indicators for the xStats plugin to poll.

To access the xStats Log Viewer from the navigation bar, click the **Administration** menu, select **Monitoring Configuration**, and then select **xStats Log Viewer**.



52.1 xStats Log

The Sources section and the Source Files section display the xStats log data for the device you select from the Devices drop-down list.

Click the Devices drop-down list and select a device. The Devices selection list includes the devices that your xStats sources discover and add to the *Device* Manager.

52.2 Sources

When you create an xStats source on the xStats Source Manager, the Sources section displays the following information.

- **Name** - Displays the xStats source name.
- **IP Address** - Displays the IP address of the device from which the source retrieves/receives xStats data.
- **Last Successful Fetch** - Displays the date and time the source most recently performed a successful collection of xStats data.
- **Last Attempted Fetch** - Displays the date and time the source most recently attempted to fetch xStats data.

52.3 Source Files

The Source Files section displays the following information.

- **Input File** - Displays the name of the file the xStats source fetched/received.
- **Fetched Time** - Displays the time the source fetched the file.

53 Processes and Logs

53.1 SevOne NMS Processes

The Cluster Manager, at the Appliance level, the Process Overview tab displays the following list of processes SevOne NMS runs.

* Processes with an asterisk (*) in the list and the following two processes (that do not appear in the list) can be monitored via the SevOne-NMS-MIB. This provides the ability to self monitor the SevOne NMS appliance. Self monitor statistics are available in Performance Metrics report attachments and Instant Graphs when you select the localhost device and its associated indicators on which to run reports.

- SevOne-clusterd - Synchronizes communication among the appliances in a clustered environment.
- SevOne-searchd - Indexes search results on the back end for the cluster wide advanced/global searches.

Processes are grouped into subsections. Some provide **Stop, Start**, and **Restart** buttons to enable you to stop and start processes. This is mainly for SevOne Support Engineers and you should not click these buttons without a strong cause.

53.1.1 Core Processes

- *MySQL Config Server - Synchronizes the Config databases of clustered appliances.
- *MySQL Data Server - Synchronizes the Data database of clustered environments.
- nginx - Displays the SevOne NMS web user interfaces.
- PHP-FPM - PHP handler used to run PHP scripts in the SevOne NMS GUI.
- SSH Daemon - Secure shell used for remote login to operate network services securely on an unsecured network.
- Syslog - Manages the logs that appear on the System Logs tab.

53.1.2 Optional Processes

- HTTP Proxy - Forwards VMware data among peers.
- sFlow Converter - Enables the NetFlow Collector process to receive sFlow data.
- Traffic samplicator (port 9997) - Listens on UDP port 9997 and replicates the traffic to any number of IP addresses and ports, similar to a broadcaster.

> ⚠ By default, the samplicator is *disabled*. For the samplicator to persist as enabled across reboots and restarts of *supervisord*, using a text editor of your choice, edit **/etc/supervisord.d/samplicator_9997.ini** file.
>
> **Edit /etc/supervisord.d/samplicator_9997.ini file to set 'autostart' to 'true'.**
>
> ```
> $ vi /etc/supervisord.d/samplicator_9997.ini
> ...
> ...
> autostart=true
> ...
> ...
> ```
>
> After updating **/etc/supervisord.d/samplicator_9997.ini**, execute the following commands to update *supervisorctl* and to start/restart the service.
>
> ```
> $ supervisorctl reread
> $ supervisorctl update
> $ supervisorctl restart samplicator_9997
> ```

For details on samplicator, please refer to section Samplicator below.

> ⚠ When you execute the following command to perform the *services* check, samplicator service is ignored and the check does not inform the user whether or not the samplicator is running.

```
$ SevOne-act check services
```

53.1.3 SevOne Daemons

- SevOne Backfill Daemon - Processes and inserts xStats data that is not current.
- *SevOne Data Daemon - Inserts polled data into the database.
- SevOne FlowDB Daemon - Processes raw NetFlow data.
- SevOne Master/Slave (Leader/Follower) Monitor - Coordinates actions between the appliances in a Hot Standby Appliance peer pair
- SevOne Message Aggregator - Handles alert messages.
- *SevOne MIB Daemon - Caches MIB information and resolves OID numbers to text.
- SevOne NetFlow Collector - Handles the NetFlow poller.
- *SevOne Realtime Scheduler - Performs high frequency polling.
- *SevOne Request Daemon - Communicates with clustered appliances to collect data for reports.
- *SevOne Scheduler - Handles poll schedules.
- SevOne Stats - Processes data for self monitoring.
- *SevOne Trap Collector - Receives and processes SNMP traps.
- *SevOne xStats Backfill Insertion Daemon - Inserts xStats data into long term tables.
- *SevOne xStats Dispatch Daemon - Performs name to ID resolution for xStats devices and associates xStats data to the correct peer.
- *SevOne xStats File Collector - Reads and parses xStats files.
- SevOne xStats Ingestion Resolver Daemon - Performs name to ID resolution for xStats objects and indicators.

53.1.4 SevOne Master/Slave (Leader/Follower) Actions

- Action: become *leader* - Makes the passive appliance in a Hot Standby Appliance peer pair take over and become the active appliance.
- Action: become *follower* - Makes the active appliance in a Hot Standby Appliance peer pair fail over and become the passive appliance.
- Action: format *follower* - Formats the database on the passive appliance in a Hot Standby Appliance peer pair.

53.1.5 SevOne Scripts

- SevOne Alert Mailer - Emails alerts that are configured to be emailed when the threshold is triggered.
- SevOne Device Mover - Processes and performs device moves from peer to peer that are initiated from the Device Mover.
- SevOne Discover Script - Handles all device discovery.
- SevOne Longterm Trim - Trims historical data based on the Cluster Manager setting.
- SevOne Report Mailer - Emails reports that are scheduled to be emailed.
- SevOne Shortterm Backup - Backs up short term data that is stored in memory.

53.1.6 SevOne Utilities

- SevOne Longterm Cacher - Caches and processes baseline data.
- SevOne Longterm Updater - Writes data stored in short term memory to the disk every two hours.
- SevOne Shortterm Trimmer - Trims short term data.
- SevOne Threshold Checker - Manages alerts.

53.2 SevOne NMS Appliance Logs

The Cluster Manager, at the Appliance level, the System Logs tab enables you to review the log files to which SevOne NMS writes.

53.2.1 System

- messages - Displays the generic log for all un-grouped messages.
- kern - Displays output of command */usr/bin/dmesg*, which prints the kernel ring buffer.

53.2.2 Script Logs

- SevOne-backup-config-data.log
- SevOne-backup.log
- SevOne-checkmate.log
- SevOne-device-mover.log - Displays the log of the devices that you move between the peers in the cluster
- SevOne-devices-deletion-queue.log - Displays the log of the devices that are added to the deletion queue to be deleted.
- SevOne-ffupdater.log
- SevOne-generate-admin-messages.log
- SevOne-mib-synchronize.log
- SevOne-summary-table-tool.log
- SevOne-tablecacher.log
- SevOne-top-highpolld.log
- SevOne-top-polld.log
- aggregated-netflow-rollup.log - Displays the log that states when the daily data points for aggregated flow data ran
- alertmailer.log - Displays the log of the script that emails new alerts. Look here when alert emails are not sent or received
- cacher.log
- checksshd.log
- disableUsers.log
- discover-netflow.log
- discover-schedule.log
- discover-thereshold.log - This utility runs frequently throughout the day and sends new alerts to the messageswitch daemon
- discover.log
- highfreqpoller.update-by-time.log - Displays the log of the high frequency poller to inform you of the poll status
- ipmi-message.log
- mailreports.log - Displays the log of the script that emails reports. Look here when report emails are not sent or received
- mysql-replication-maintainer-config.log - Displays logs for MySQL config database
- mysql-replication-maintainer-data.log - Displays the logs for the MySQL data database
- mysqloptimize_config.log
- mysqloptimize_data.log
- namflow.log
- periodic.shortterm.backup.log - Displays the log for the short term backup. This utility writes the status of the periodic memory-table backups that are made when a server reboots
- proxy-write-config.log
- rapid-plugins-pdf.log
- rest-api-keepalive.log
- sevone-cert-update.log
- snmpd-restart.log
- sync-ldap-groups.log
- sysuptime-normalize.log
- trim-alerts.log
- trim-bulkdlogs.log
- trim-device.log
- trim-longterm.log
- trim-mysqllogs.log
- trim-netflow.log
- trim-netflowaggregate.log
- trim-rtagx.log
- trim-sessions.log
- trim-shortterm-netflow.log
- trim-shortterm.log
- trim-shorttermaggregate-daily.log
- trim-shorttermaggregate-hourly.log
- trim-shorttermaggregate-monthly.log
- trim-shorttermaggregate-sixhourly.log
- trim-shorttermaggregate-weekly.log
- trim-sixhourlynetflow.log
- trim-temporarytable.log
- trim-traps.log - Displays the log of the traps received. The output is in hexadecimal format
- updater.log - Displays the logs for the hourly, daily, weekly, monthly, quarterly, and yearly updater. This utility writes short term memory data to disk
- updateraggregate.daily.log

- updateraggregate.monthly.log
- updateraggregate.sixhourly.log
- updateraggregate.weekly.log
- upgrade-appliance.log
- vcenterupdate.log
- write-ldap-certs.log

53.2.3 Other Logs

- /var/date.log

53.2.4 General Logs

- SevOne-audit.log
- SevOne-clusterd.log
- SevOne-datad.log
- SevOne-device-scand.log
- ingestion/SevOne-dispatchd.log
- ingestion/SevOne-fcad.log
- SevOne-flowdb.log
- SevOne-highpolld.log
- ingestion/SevOne-ingestion-resolved.log
- SevOne-insert-backfild.log
- SevOne-masterslaved.log - Displays information about the active appliance in a Hot Standby Appliance peer pair and its relationship with the passive appliance in the peer pair
- SevOne-mibd.log
- SevOne-netflow-cleanup.log - Displays the NetFlow daemon updater process
- SevOne-polld.log
- SevOne-requestd.log
- SevOne-searchd.log
- SevOne-statsd.log
- SevOne-topologyd.log
- SevOne-trapd.log - Displays the trap daemon
- cron.log
- discovery.log - Displays the log of the regular discovery script, which runs frequently throughout the day to discover device updates and new devices
- logrotate.log
- messageswitch.log - Displays the log of the alert handler. All new alerts come through this system
- mysql/mysqld.err
- mysql/mysql2.err
- mysql/mysqld_multi.log
- net-snmpd.log
- nginx.err
- nginx.log
- nginx/access.log
- nginx/error.log
- php-fpm.err
- php-fpm.log
- php-fpm/error.log
- php-fpm/www-error.log
- rest-api/SevOne-rest-api.log
- sftp.log
- sshd.log
- tacc.log
- trim.log
- xStats-parsers.log
- xstats/ALU5620SAMTransform/adapter.log
- xstats/AwsTransform/adapter.log
- xstats/CanaryTransform/adapter.log

53.3 Samplicator

Samplicator is a UDP datagram forwarding program. In SevOne NMS, it is most commonly used to forward NetFlow data to a different port or onto other systems. It can be used to forward any UDP data.

Assume that there is NetFlow source (device IP address 10.0.0.61) coming in on port 1234 and it cannot be changed on the device. The flow must be rerouted to the standard port, 9996, for SevOne NMS to process it normally. Please refer to section Configure Samplicator below for details.

53.3.1 Configure Samplicator

1. Copy the samplicator example configuration file, **/etc/conf.d/samplicator.example.confd**, to **/etc/conf.d/samplicator.1234.confd**.

   ⚠ Port **1234** is being used as an example here on which this instance of the samplicator service is being configured for and will listen on.

   ```
   $ cp /etc/conf.d/samplicator.example.confd /etc/conf.d/samplicator.1234.confd
   ```

2. Using a text edit of your choice, edit **/etc/conf.d/samplicator.1234.confd** file to add device IP address 10.0.0.61 and save it.

   Example

   ```
   $ vi /etc/conf.d/samplicator.1234.confd

   #Config file format:
   #a.b.c.d[/e.f.g.h]: receiver ...
   #where:
   #a.b.c.d     is the sender's IP address
   #e.f.g.h     is a mask to apply to the sender (default 255.255.255.255)
   #receiver    see above.
   10.0.0.61: 10.0.0.60/9996
   ```

   ⓘ **/etc/conf.d/samplicator.1234.confd file format...**

   The first column is where the samplicator expects the UDP packets from. i.e., *10.0.0.61*, as shown in the example above.
   The second column is where the UDP packets go to followed by the port number. i.e., *10.0.0.60/9996*, as shown in the example above.
   Alternatively, you may use 0.0.0.0 as the receiver IP address if it does not matter where the data is coming from and you want to forward all UDP traffic coming in on the specified port.

3. Now that the configuration file has been modified, set the parameters for **supervisord** to start the samplicator for this specific port forwarding request.

   ⚠ Samplicator configurations can be created individually, or many different incoming / outgoing pairs can configured for the same samplicator instance. This depends on the requirements of the environment.

4. Copy **/etc/supervisord.d/samplicator_9997.ini** file to **/etc/supervisord.d.<***master or dnc or slave folder***>/samplicator_1234.ini**. Please see the note below to determine whether to copy the file in *master or dnc* or *slave* folder.

> **Master / Leader or any other role...**
>
> If the samplicator service needs to be running when the appliance is in *Master* / *Leader* state, copy the file to **/etc/supervisord.d.master** folder.
> For an active *DNC* appliance, copy the file to **/etc/supervisord.d.dnc** folder.
> In any other role, copy to **/etc/supervisord.d.slave** folder.
> For the steps below, we will assume that the appliance is in **Master** / **Leader** state.

Example

```
$ cp /etc/supervisord.d/samplicator_9997.ini /etc/supervisord.d.master/samplicator_1234.ini
```

5. Using a text edit of your choice, edit **/etc/supervisord.d.master/samplicator_1234.ini** file to update the **program** name to reflect the samplicator port being used. i.e., *[program:samplicator_1234]*. Also, update the **command** to reflect the configuration file name (using option **-c**) and the port (using option **-p**). After the updates, save the file.

Example

```
$ vi /etc/supervisord.d.master/samplicator.1234.ini

[program:samplicator_1234]
command=/usr/bin/samplicate -S -c /etc/conf.d/samplicator.1234.confd -p 1234 -d0
stdout_logfile=/var/log/samplicator.log
stderr_logfile=/var/log/samplicator.err
priority=500
autostart=true
startsecs=10
startretries=10000
autorestart=true
```

You may leave **stdout_logfile** and **stderr_logfile** as-is if you want to send the output to the same log file as the other samplicator instances. Or, you may also choose to give it a separate log file name.

6. After configuring **/etc/conf.d/samplicator.1234.confd** and **/etc/supervisord.d.master/samplicator_1234.ini** files, execute the following command.

```
$ supervisorctl reread && supervisorctl update

samplicator_1234: available
samplicator_9997: changed
samplicator_9997: stopped
samplicator_9997: updated process group
samplicator_1234: added process group
```

7. Start the samplicator process for the new configuration.

```
$ supervisorctl start samplicator_1234
```

53.3.2 Test Samplicator

You may now run *tcpdump* on the destination port you have configured above to ensure that the samplicator is working properly. If you do not have any data coming in, you may send the test data to the samplicator port on the appliance.

Example

```
$ echo "Hello World" >/dev/udp/127.0.0.1/1234
```

53.3.3 Some Considerations

Ensure that the port you have chosen is not already in-use for the samplicator to listen on. If another process is bound to the port, the samplicator will fail to start.

53.3.3.1 Autostart

In order to ensure that the samplicator starts on boot, the **autostart** line in **/etc/supervisord.d.master/samplicator_1234.ini** file must be set to **true**. By default, it is set to **false**, and without changing this setting, samplicator will not start on reboot despite the presence of it in */etc/supervisord.d.master/samplicator_1234.ini* file.

```
autostart=true
```

53.3.3.2 Configuration

The configuration files managed by the **supervisord** daemon can be found in **/etc/supervisord.d** directory. SevOne NMS maintains different **supervisord** startup configuration **.ini* files placed within the directories that are relevant to the role (*master*, *slave*, or *dnc*) of the appliance in SevOne NMS. For example, all services that need to be configured to run on an *active* **PAS** are placed in **/etc/supervisord.d.master**. For an *active* **DNC** appliance, the files are placed in **/etc/supervisord.d.dnc**. For all appliances with a *passive* role (PAS or DNC) are placed in **/etc/supervisord.d.slave**.

When an appliance assumes an *active* or *passive* role, SevOne NMS creates a symbolic link for **/etc/supervisord.d** to point to the relevant directory based on its role in SevOne NMS. Creating the samplicator configuration file in the correct folder (*/etc/supervisord.d.master*, */etc/supervisord.d.slave*, or */etc/supervisord.d.dnc*) is important.

> **Example**
>
> If you have a samplicator service running when the DNC appliance is in *active* state, the samplicator file must be created in */etc/supervisord.d.dnc*. In case of a failover / takeover, if the DNC appliance assumes a passive role, the samplicator service will no longer run on that DNC. You will need to create the samplicator file in */etc/supervisord.d.slave* directory for the samplicator service to continue to run after a failover / takeover.

NMS Role	Samplicator Configuration File Location
PAS *active*	/etc/supervisord.d.master
PAS *passive*	/etc/supervisord.d.slave
DNC *active*	/etc/supervisord.d.dnc
DNC *passive*	/etc/supervisord.d.slave

The same configuration file must be updated in the relevant directories on the *primary* and *secondary* appliance of the pair to ensure that the samplicator service continues to run on the appliance after a failover / takeover.

For additional details, please refer to https://github.com/sleinen/samplicator/.

54 Trap Revisions

SevOne NMS provides trap revisions - one, three, and four. The Cluster Manager > **Cluster Settings** tab > **Alerts** subtab, enables you to select which trap revision to use. If you change the trap revision you will need to update how your fault management system receives traps from SevOne NMS.

54.1 Revision One

Revision One traps provide the following fields.

54.1.1 Core Message

- s1TrapHistoryThresholdR1Severity
- s1TrapHistoryThresholdR1Action
- s1TrapHistoryThresholdR1Type
- s1TrapHistoryThresholdR1GmtTime
- s1TrapHistoryThresholdR1UserName
- s1TrapHistoryThresholdR1MessageText

54.1.2 Peer Information

- s1TrapHistoryThresholdR1PeerName
- s1TrapHistoryThresholdR1PeerId
- s1TrapHistoryThresholdR1PeerIp

54.1.3 Device Information

- s1TrapHistoryThresholdR1DeviceName
- s1TrapHistoryThresholdR1DeviceId
- s1TrapHistoryThresholdR1DeviceIp

54.1.4 Threshold Information

- s1TrapHistoryThresholdR2ThresholdName
- s1TrapHistoryThresholdR2ThresholdDesc
- s1TrapHistoryThresholdR2DeviceGroupID
- s1TrapHistoryThresholdR2DeviceGroupName
- s1TrapHistoryThresholdR2ObjectGroupID
- s1TrapHistoryThresholdR2ObjectGroupName
- s1TrapHistoryThresholdR2DeviceAltName

54.1.5 Plugin Information

- s1TrapHistoryThresholdR1PluginName
- s1TrapHistoryThresholdR1PluginId
- s1TrapHistoryThresholdR1PluginDescription

54.1.6 Object Information

- s1TrapHistoryThresholdR1ObjectName
- s1TrapHistoryThresholdR1ObjectId
- s1TrapHistoryThresholdR1ObjectDescription

54.1.7 Indicator Information

- s1TrapHistoryThresholdR1IndicatorName
- s1TrapHistoryThresholdR1IndicatorId
- s1TrapHistoryThresholdR1IndicatorDescription

54.1.8 Condition Information

- s1TrapHistoryThresholdR2ConditionID
- s1TrapHistoryThresholdR2ConditionType
- s1TrapHistoryThresholdR2ConditionValue
- s1TrapHistoryThresholdR2ComparisonValue
- s1TrapHistoryThresholdR2ConditionUnits
- s1TrapHistoryThresholdR2ConditionAggregation
- s1TrapHistoryThresholdR2ConditionTimeframe

54.2 Revision Three

Revision Three traps provide the fields listed above in addition to the following fields.

> s1TrapHistoryThresholdR1PeerIp and s1TrapHistoryThresholdR1DeviceIp are always blank when using Revision Three traps.

- s1TrapHistoryThresholdR3PeerAddress
- s1TrapHistoryThresholdR3DeviceAddress
- s1TrapHistoryThresholdR3PolicyID
- s1TrapHistoryThresholdR3ThresholdID
- s1TrapHistoryThresholdR3ThresholdDescription

54.3 Revision Four

Revision Four traps has removed the following fields from the new **sevoneTrapNotificationThresholdRevision4**. By doing this, SNMP/Other traps can be sent from SevOne NMS with fewer **varbinds** so that the sent traps have a lower chance of exceeding the network interface MTU.

- s1TrapHistoryThresholdR1Index
- s1TrapHistoryThresholdR1PeerId
- s1TrapHistoryThresholdR3PeerAddress
- s1TrapHistoryThresholdR2DeviceAltName
- s1TrapHistoryThresholdR1PluginDescription
- s1TrapHistoryThresholdR1IndicatorDescription
- s1TrapHistoryThresholdR2ConditionID
- s1TrapHistoryThresholdR2ConditionType
- s1TrapHistoryThresholdR2ConditionValue
- s1TrapHistoryThresholdR2ComparisonValue
- s1TrapHistoryThresholdR2ConditionUnits
- s1TrapHistoryThresholdR2ConditionAggregation
- s1TrapHistoryThresholdR2ConditionTimeframe

For the **Flow** traps, the following fields have been removed from the new **sevoneTrapNotificationFlowThresholdRevision2**. By doing this, Flow traps can be sent from SevOne NMS with fewer **varbinds** so that the sent traps have a lower chance of exceeding the network interface MTU.

- s1TrapHistoryThresholdR1PeerId
- s1TrapHistoryThresholdR3PeerAddress
- s1TrapHistoryThresholdR2DeviceAltName
- s1TrapHistoryThresholdR1PluginDescription
- s1TrapHistoryThresholdR1IndicatorDescription
- s1TrapHistoryThresholdR2ConditionID
- s1TrapHistoryThresholdR2ConditionType
- s1TrapHistoryThresholdR2ConditionValue
- s1TrapHistoryThresholdR2ComparisonValue
- s1TrapHistoryThresholdR2ConditionAggregation
- s1TrapHistoryThresholdR2ConditionTimeframe
- sevoneTrapHistoryFlowThresholdR1ConditionElementId

55 Perl Regular Expressions

55.1 Regexes

Regular expressions (commonly called regexes) are tools used for pattern matching. They are useful to find the answer to questions like, "Is this text like such-and-such", or for queries like, "Find me all items like this-and-that".

SevOne NMS uses regular expressions throughout the application and many fields enable you to enter your own regular expressions.

55.2 Regular Expressions

Each kind of regular expression has its own style and rules. For example, DOS-style regular expressions perform wildcard matching:

Character	Meaning
`*`	Matches any number of characters (including zero)
`?`	Matches any one character

And SQL supports the following wildcards:

Character	Meaning
`%`	Matches any number of characters (including zero)
`_`	Matches any one character

BASH regular expressions are more powerful. Linux supports both * and ? plus the idea of a character class.

Character	Meaning
*	Matches any number of characters (including zero)
?	Matches any one character
`[a-zA-Z]`	Matches any English letter

55.3 Perl Regular Expressions

Some of the most widely used regular expressions are Perl regular expressions that were used internally by the Perl scripting language. Perl regular expressions use a combination of normal and special characters to define match statements. By default, they operate on one line of text only. A line is delimited in one of two ways: by simply terminating; or by a carriage return or line feed combination (\r) and (\n).

Character	Meaning
.	Matches any single character
`^`	Matches the beginning of the string
`$`	Matches the end of the string

Character	Meaning
`character`	Matches a specific character
`[characters]`	Matches the class of characters or ranges of characters within the square brackets
`|`	Delimiter between two strings to match
`()`	Used to group a collection of items
{x,y}	To match between the xth and yth characters of a string. If the second number is left out, no limit is imposed
`?`	To match one or none of something. This is equivalent to `{0,1}`.
`+`	To match one or more of something. This is equivalent to {1,}.
`*`	To match any or none of something. This is equivalent to {0,}.
`\`	To match any of the special characters listed, you may prefix that character with a \. Also known as 'escaping' characters.

55.3.1 Examples

For a router with the name of RTR NYC 01, you could use the following to match that string exactly. This also matches Wireless RTR NYC 01.

```
RTR NYC 01
```

To match only the Wireless RTR NYC 01 string, you could use:

```
^Wireless RTR NYC 01$
```

To match any string that begins with Wireless, you could use:

```
^Wireless
```

To match all strings with Wireless or RTR in them, you could use:

```
Wireless|RTR
```

To match any IP address, you could use:

```
[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}
```

To match a DNS name, you could use:

```
([a-zA-Z0-9][a-zA-Z0-9-]*\.)+[a-zA-Z0-9-]
```

56 Glossary and Concepts

56.1 Glossary

The Glossary explains SevOne NMS vocabulary.

- Active Appliance - The SevOne NMS appliance in a Hot Standby Appliance (HSA) peer pair that actively polls, alerts, and reports. Upon initial setup the primary appliance is the active appliance in the peer pair. If the primary appliance fails, the secondary appliance becomes the active appliance.
- Aggregation - Enables you to manipulate the granularity of the data points in graphs and to define how to calculate each data point in order to smooth a graph over the time span you define.
- Alerts - Current, active messages include threshold violations, trap notifications, and website errors. Alerts you manually acknowledge or are cleared with a clear condition appear on the Alert Archives page.
- Appliance - The hardware on which the SevOne NMS software runs. In your cluster each appliance can be a peer. When your cluster includes a Hot Standby Appliance (HSA) peer pair, there are two appliances that act as one peer to provide redundancy. If the primary appliance fails, the secondary appliance becomes the active appliance.
- AWS - This acronym stands for Amazon Web Services.
- Baselines – Default baseline granularity takes all data points in a 15 minute time span, averages them, and stores that average for every 15 minutes of the week for a total of 672 data points. The Baseline Rule Manager enables you to create rules to enable or disable baselines. The Reset Baselines page enables you to reset the baseline values for the time span you specify.
- Candidate - A candidate is something that a network scan successfully pings. A candidate has not been added into SevOne NMS and is not polled for metrics. In order to poll metrics for reports and alerts, you must add a candidate into SevOne NMS where it becomes a device.
- Chain Reports - The ability to use the settings from one attachment to create a related attachment that drills down to more specific data or provides related data for the same set of devices, objects, interfaces, etc.
- Cluster - An interconnected set of SevOne appliances that exchange information about the network devices from which they collect statistical data.
- Cluster Leader – The SevOne NMS peer that stores the master copy of the Cluster Manager settings, security settings, and other global settings. All other active peers in your SevOne NMS cluster pull the data from the cluster leader peer config database.
- Device - A device is composed of a collection of objects that represents a self-contained entity of some kind.
- Device Discovery - The process to query and update information about the devices that are in SevOne NMS. The manual discovery process runs every two minutes to test the various plugins/technologies only on the devices you mark for discovery. The automatic discovery process runs on a schedule (usually daily) to test the various plugins/technologies on all devices in SevOne NMS. Device discovery creates new objects in SevOne NMS, updates existing objects, and ultimately deactivates and deletes unused objects.
- Device Groups – Enable you to organize devices for reports and security purposes.
- Device Manager - Displays the devices in SevOne NMS to which you have permissions. This page enables authorized users to add, edit, and delete devices and to manage device plugins, polling, and discovery.
- Device Mover - Enables you to move devices from one SevOne NMS peer to another SevOne NMS peer.
- Device Summary - Displays device specific statistics from the ICMP plugin, Process plugin, Databases plugin, SNMP plugin, and VMware plugin plus a list of the alerts for the device.
- Device Type – Enables you to organize devices for SNMP polling purposes. You can view devices as members of a device type similarly to the relationship that many individual objects can be viewed as if they all belong to one platonic object type. A device type is more flexible than an object type.
- FlowFalcon - The SevOne NMS flow collector for flow technologies such as NetFlow. The flow report suite is known as FlowFalcon.
- High Frequency Poller - Enables you to poll individual objects on a device faster than the standard once per minute. This helps you detect spikes in network traffic that last less than a few seconds.
- Hot Standby Appliance (HSA) - A complete mirror of the Cluster Leader or any other peer appliance in your SevOne NMS cluster.
- Indicator - Object level metrics are called indicators. An object represents a logical entity that is some part of the device which can provide metrics about itself.
- Instant Graphs – Provide a quick and easy way to view the status and performance of your network's devices, objects, and indicators.
- IP SLA – This acronym stands for Internet Protocol Service Level Agreement. IP SLAs enable you to monitor the network performance between two Cisco routers. IP SLA is a feature that is embedded in the Cisco IOS software that SevOne NMS can monitor to help Cisco customers understand IP service levels, increase productivity, lower operational costs, and reduce the frequency of network outages.
- Neighbor - The other appliance in a Hot Standby Appliance peer pair. The primary appliance's neighbor is the secondary appliance and vice versa.

- NMS - This acronym stands for Network Management System.
- Object - An object or element is a discrete component of a device or a software component that has one or more performance indicators that can be monitored, trended, or alerted on. In SevOne NMS, an element is considered any performance object.
- PAS - This acronym stands for Performance Appliance Solution.
- Passive Appliance - The SevOne NMS appliance in a Hot Standby Appliance peer pair that replicates the databases of the active peer appliance. Upon initial setup the secondary appliance is the passive appliance.
- Peer – Each SevOne NMS appliance in your implementation is either a peer within your SevOne NMS cluster or the Hot Standby Appliance to the active appliance in a Hot Standby Appliance peer pair. Each active peer pulls a full replica of the cluster leader peer configuration database and maintains the performance data for the devices it polls. Your cluster can peer SevOne NMS PAS appliances and SevOne NMS DNC appliances and can include Hot Standby Appliance peer pairs.
- Pin - To manually add a device to a device group/device type or to manually add an object to an object group in such a way that it cannot be removed from the device group/device type/object group via rule or discovery. You must manually unpin a pinned device/object to remove the device/object from the device group/device type/object group.
- Plugin – The SevOne NMS mechanisms that poll (collect, ask for, etc.) data from technologies. A plugin defines the following:
    - A way to get data - Usually via some protocol such as SNMP, ICMP, WMI, etc.
    - Object Types - Define logical things to ask for information about.
    - Indicator Types - Define kinds of metrics that object types can have.
- Policy - The framework that enables you to define a threshold to apply for a device group/device type. A threshold is the value that triggers an alert or a trap.
- Poll - The process of using the plugins you enable on a device to gather the metrics on which SevOne NMS can generate reports and alerts.
- Portshaker - The Portshaker plugin enables you to check whether the device is listening on a specific TCP port as well as graph its response time.
- Primary - The appliance in a Hot Standby Appliance peer pair that is initially configured to be the active, normal, polling appliance. If the primary appliance fails, it is still the primary appliance but it becomes the passive appliance.
- Process - The Process plugin enables you to collect performance and availability information about individual processes running on a device.
- Proxy Ping – The Proxy Ping plugin enables SevOne NMS to have a router ping another router to find the latency of a link.
- Remote Plugin Manager - Remote plugin managers enable the placement of a SevOne collector closer to the devices to monitor. This enables collection from within a network via Network Address Translation and can reduce network traffic over a bandwidth limited WAN. Like other plugins, remote plugin managers discover and monitor devices via the protocols that the remote plugin manager is designed to leverage.
- Report Template - Report Templates are similar to reports with the added ability to define template attachments that do not have a specific resource. You define the report template properties to enable applicable template attachments to derive their device resources from the Device Summary workflows or to derive their object resources from the Object Summary workflows. Report templates enable you to create a report that has template attachments without a specific resource and attachments with specific resources.
- Secondary - The appliance in a Hot Standby Appliance peer pair that is initially configured to be the passive appliance. If the active appliance fails, this is still the secondary appliance but the secondary appliance assumes the active role.
- SNMP - This acronym stands for Simple Network Management Protocol. SNMP is a key technology for network management. Virtually all operating systems support SNMP. Devices that support SNMP run an agent that stores information about the device in a tree-like structure of Object Identifiers (OIDs). SevOne NMS displays OIDs as their corresponding Management Information Bases (MIBs). Devices send SevOne NMS SNMP traps and SevOne NMS can send traps to other trap destinations.
- Threshold – The value that triggers an alert or trap. The Threshold Browser enables you to create thresholds for an individual device and the Policy Browser enables you to define a policy which is a threshold that applies to a device group/device type.
- WMI - The WMI plugin enables you to monitor Windows Management Instrumentation statistics.

56.2 Concepts

SevOne deploys as a physical or virtual appliance. A single SevOne appliance monitors up to 200,000 objects. You can peer appliances together into a cluster to increase monitoring capacity. Each appliance you peer into your cluster collects, stores, and reports metrics from the devices you assign the peer to monitor.

The peer-to-peer, cluster approach enables users to log on to any SevOne peer and view information about the entire network. When a report spans the devices from multiple peers, each peer works on its part of the report and sends its metrics to the peer that made the request.

The SevOne NMS application monitors your network. Your network has many metrics. SevOne NMS can scan your network to find candidates. When you add candidates to SevOne NMS as a device, technology specific plugins discover the objects that are members of technology specific object types on the device. The plugin then polls those objects to gather metrics from the indicators that are

contained in the object type specific indicator types. You can choose to turn on the plugins you deem relevant to gather metrics from the technologies that matter to you.

From the opposite perspective: Metrics are polled from indicators. Indicators are grouped into technology specific indicator types. Indicator types are conceptually grouped into object types. Each object type groups objects by technology. Objects are physical or virtual parts of a device that contain the indicators that generate metrics.

There are two ways to organizing devices. The typical SevOne NMS user with report view and alert management permissions will note that SevOne NMS treats both device groups and device types similarly.

- Device Groups enable you to organize devices into logical entities for security, report, and alert purposes. A user with permissions to manage devices can manage device groups but cannot manage device types.
- Device Types enable you to organize devices into technological entities based on the discovery of similar SNMP objects. Device type management is restricted to more administrative users because device types have additional device discovery aspects.

56.2.1 Candidate

A candidate is something that a network scan successfully pings. A candidate has not been added into SevOne NMS and is not polled for metrics. In order to poll metrics for reports and alerts, you must add a candidate into SevOne NMS where it becomes a device.

56.2.2 Device

A device is composed of a collection of objects that represents a self-contained entity of some kind.

- Desktop Computer
- Server in the Datacenter
- Network Router
- Network Switch
- Network Firewall
- Load Balancer
- Car
- House

56.2.3 Object

Each object is a part of a device. The relationship is deliberate and is not subject to change. An object represents a logical entity that is some part of the device which can provide metrics about itself. Object level metrics are called indicators. In the examples, an object is either a component of the device or an object represents some logical entity that makes sense within the context of the device.

- Device - Desktop Computer
    - Object - Ethernet Port
    - Object - First Hard Drive
- Device - Server in the Datacenter
    - Object - First Ethernet Port
    - Object - First RAID Array
- Device - Network Router
    - Object - First Ethernet Port
    - Object - Routing Processor
- Device - Network Switch
    - Object - First Ethernet Port
    - Object - Switching Processor
- Device - Network Firewall
    - Object - First Ethernet Port
    - Object - Processor
- Device - Load Balancer
    - Object - First Ethernet Port
    - Object - Site that is being load-balanced
- Device - Car
    - Object - Driver Side Tire
    - Object - Main Processor
- Device - House

- Object - Smoke Alarm
- Object - Thermostat

56.2.4 Object Type

Many devices in the examples have an Ethernet port object. These Ethernet ports are somehow related. They are all the same because they are all Ethernet ports. They are all different because each Ethernet port is physically distinct from the other Ethernet ports. This abstraction of like objects is called an object type. All objects must have an object type. The object type describes objects as a concept, outside of their individual devices. An object can belong to only one object type.
Object types can be further abstracted and the more abstract form of an object type is simply another object type. This generalization could continue indefinitely and hierarchically. The collection of all object types is called the object type hierarchy. Object types can be grouped hierarchically by plugin. This enables object abstraction. All objects within each hierarchical grouping are treated equally (e.g., CPU), irrespective of collection method, which make it much easier to define thresholds and to create reports.

56.2.5 Indicator

Object level metrics are called indicators. Remember, an object represents a logical entity that is some part of the device which can provide metrics about itself.

56.2.6 Indicator Type

All indicators of an object must have an indicator type. This means that an object has indicators and an object type has indicator types. An actual object that provides actual indicators is a specific instance of an object type that has indicator types.

56.2.7 Device Type

You can view devices as members of a device type similarly to the relationship that many individual objects can be viewed as if they all belong to one platonic object type. A device type is more flexible than an object type. Device types enable you to use SNMP discovery to organize the polled metrics for reports and alerts. For example: A specific device could apply to each of the following:

- A Computer
- A Server
- A Linux Box
- A Web Host
- A BitCoin Miner
- A Database

All of these statements can be simultaneously true. Each of these statements about devices enables you to determine the type of things to consider and each could be a device type. A device can be a member of many device types. A device is a member of a device type and a device type contains that device. The separation of device types is dependent on what you expect to see when you look at the device. For example, here are the things that you might want to see for each device type:

- A Computer
    - CPU
    - Hard Drive
    - Memory
- A Server
    - RAID Array
    - Ethernet Port
- A Linux Box
    - Kernel Status
    - Users
    - Quota
- A Web Host
    - Apache Service
    - Web Sites
- A BitCoin Miner
    - Miner Processes
    - Bank Accounts
- A Database
    - MySQL Service

- Oracle Services

A device type is primarily defined by its list of distinguishing object types. The things you expect to see are the defining characteristics of each device type. Any one of the things listed for one device type could be present for any other device type. However, each defining characteristic is listed under the most direct device type, the one that is most defined by those things. Device types may share object types with other device types.
The collection of device types and all of their associated object types is called the device type hierarchy. SevOne NMS supports a device type hierarchy that can extend more than twenty levels.
Users need administrative permissions to manage device types. Users only need the Can Manage Devices permissions to manage device groups.

56.2.8 Device Group

You can organize devices into device groups. Device groups are more flexible than device types and are generally based on factors such as device location, accessibility, manufacturer, function, etc. As an Internet Service Provider (ISP) you could group devices by customer. You could group each customer's network devices into separate device groups to prevent your other customer from seeing metrics polled from their competitor's network devices. You could work your way down the device group hierarchy to further group devices into regions, etc.
Users with Can Manage Devices permission can manage device groups but cannot manage device types.