



Configuring SNMP

This chapter describes how to configure the SNMP feature on Cisco NX-OS devices.

This chapter contains the following sections:

- [About SNMP, on page 1](#)
- [Guidelines and Limitations for SNMP, on page 7](#)
- [Default Settings for SNMP, on page 8](#)
- [Configuring SNMP, on page 8](#)
- [Configuring the SNMP Local Engine ID, on page 31](#)
- [Verifying SNMP Configuration, on page 32](#)
- [Configuration Examples for SNMP, on page 33](#)
- [Additional References, on page 35](#)

About SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

SNMP Functional Overview

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The Cisco Nexus device supports the agent and MIB. To enable the SNMP agent, you must define the relationship between the manager and the agent.
- A managed information base (MIB)—The collection of managed objects on the SNMP agent

SNMP is defined in RFCs 3411 to 3418.

The device supports SNMPv1, SNMPv2c, and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security.

Cisco NX-OS supports SNMP over IPv6.

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Cisco NX-OS generates SNMP notifications as either traps or informs. A trap is an asynchronous, unacknowledged message sent from the agent to the SNMP managers listed in the host receiver table. Informs are asynchronous messages sent from the SNMP agent to the SNMP manager which the manager must acknowledge receipt of.

Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap. The device cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If the device never receives a response, it can send the inform request again.

You can configure Cisco NX-OS to send notifications to multiple host receivers.

The following table lists the SNMP traps that are enabled by default.

Trap Type	Description
generic	: coldStart
entity	: entity_fan_status_change
entity	: entity_mib_change
entity	: entity_module_status_change
entity	: entity_module_inserted
entity	: entity_module_removed
entity	: entity_power_out_change
entity	: entity_power_status_change
entity	: entity_unrecognised_module
link	: cErrDisableInterfaceEventRev1
link	: cieLinkDown
link	: cieLinkUp
link	: cmn-mac-move-notification
link	: delayed-link-state-change
link	: extended-linkDown
link	: extended-linkUp
link	: linkDown
link	: linkUp
rf	: redundancy_framework

Trap Type	Description
license	: notify-license-expiry
license	: notify-no-license-for-feature
license	: notify-licensefile-missing
license	: notify-license-expiry-warning
upgrade	: UpgradeOpNotifyOnCompletion
upgrade	: UpgradeJobStatusNotify
entity	: entity_sensor
rmon	: fallingAlarm
rmon	: hcRisingAlarm
rmon	: hcFallingAlarm
rmon	: risingAlarm

SNMPv3

SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The security features provided in SNMPv3 are the following:

- Message integrity—Ensures that a packet has not been tampered with in-transit.
- Authentication—Determines the message is from a valid source.
- Encryption—Scrambles the packet contents to prevent it from being seen by unauthorized sources.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Security Models and Levels for SNMPv1, v2, v3

The security level determines if an SNMP message needs to be protected from disclosure and if the message needs to be authenticated. The various security levels that exist within a security model are as follows:

- noAuthNoPriv—Security level that does not provide authentication or encryption. This level is not supported for SNMPv3.
- authNoPriv—Security level that provides authentication but does not provide encryption.
- authPriv—Security level that provides both authentication and encryption.

Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. The security model combined with the security level determine the security mechanism applied when the SNMP message is processed. The following table identifies what the combinations of security models and levels mean.

Table 1: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	authNoPriv	HMAC-MD5, HMAC-SHA, or SHA-256	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5, HMAC-SHA, or SHA-256	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaining (CBC) DES (DES-56) standard.

User-Based Security Model

The SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur nonmaliciously.
- Message origin authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- Message confidentiality—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

Cisco NX-OS uses three authentication protocols for SNMPv3:

- HMAC-MD5-96 authentication protocol
- HMAC-SHA-96 authentication protocol
- SHA-256 authentication protocol

Beginning with Cisco NX-OS release 9.3(7), HMAC-SHA-256 authentication protocol is used for SNMPv3.



Note When SHA-256 SNMP users are configured on the switch, ISSD is recommended by **install all** cmd else there will be config loss.

Cisco NX-OS uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option and the **aes-128** token indicate that this privacy password is for generating a 128-bit AES key. The AES priv password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 case-sensitive, alphanumeric characters. If you use the localized key, you can specify a maximum of 130 characters.



Note For an SNMPv3 operation using the external AAA server, you must use AES for the privacy protocol in the user configuration on the external AAA server.

CLI and SNMP User Synchronization

SNMPv3 user management can be centralized at the Access Authentication and Accounting (AAA) server level. This centralized user management allows the SNMP agent in Cisco NX-OS to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. Additionally, the AAA server is also used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

Cisco NX-OS synchronizes the user configuration in the following ways:

- The authentication passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes the authentication and privacy passphrases for the SNMP user.
- If you create or delete a user using either SNMP or the CLI, the user is created or deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.
- Role changes (deletions or modifications) from the CLI are synchronized to SNMP.



Note When you configure a passphrase/password in localized key/encrypted format, Cisco NX-OS does not synchronize the user information (passwords, roles, and so on).

Cisco NX-OS holds the synchronized user configuration for 60 minutes by default.

Group-Based SNMP Access



Note Because *group* is a standard SNMP term used industry-wide, we refer to roles as groups in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with read access or read-write access.

You can begin communicating with the agent once your username is created, your roles are set up by your administrator, and you are added to the roles.

SNMP and Embedded Event Manager

The Embedded Event Manager (EEM) feature monitors events, including SNMP MIB objects, and triggers an action based on these events. One of the actions could be to send an SNMP notification. EEM sends the `cEventManagerPolicyEvent` of `CISCO-EMBEDDED-EVENT-MGR-MIB` as the SNMP notification.

Multiple Instance Support

A device can support multiple instances of a logical network entity, such as protocol instances or virtual routing and forwarding (VRF) instances. Most existing MIBs cannot distinguish between these multiple logical network entities. For example, the original OSPF-MIB assumes a single protocol instance on a device, but you can now configure multiple OSPF instances on a device.

SNMPv3 uses contexts to distinguish between these multiple instances. An SNMP context is a collection of management information that you can access through the SNMP agent. A device can support multiple contexts for different logical network entities. An SNMP context allows the SNMP manager to access one of the multiple instances of a MIB module supported on the device for the different logical network entities.

Cisco NX-OS supports the `CISCO-CONTEXT-MAPPING-MIB` to map between SNMP contexts and logical network entities. You can associate an SNMP context to a VRF, protocol instance, or topology.

SNMPv3 supports contexts with the `contextName` field of the SNMPv3 PDU. You can map this `contextName` field to a particular protocol instance or VRF.

For SNMPv2c, you can map the SNMP community to a context using the `snmpCommunityContextName` MIB object in the `SNMP-COMMUNITY-MIB` (RFC 3584). You can then map this `snmpCommunityContextName` to a particular protocol instance or VRF using the `CISCO-CONTEXT-MAPPING-MIB` or the CLI.

High Availability for SNMP

Cisco NX-OS supports stateless restarts for SNMP. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Virtualization Support for SNMP

Cisco NX-OS supports one instance of the SNMP. SNMP supports multiple MIB module instances and maps them to logical network entities.

SNMP is also VRF aware. You can configure SNMP to use a particular VRF to reach the SNMP notification host receiver. You can also configure SNMP to filter notifications to an SNMP host receiver based on the VRF where the notification occurred.

Guidelines and Limitations for SNMP

SNMP has the following configuration guidelines and limitations:

- Access control list (ACLs) can be applied only to local SNMPv3 users configured on the switch. ACLs cannot be applied to remote SNMPv3 users stored on Authentication, Authorization, and Accounting (AAA) servers.
- Cisco NX-OS supports read-only access to some SNMP MIBs. See the Cisco NX-OS MIB support list at the following URL for more information: <ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html>
- Cisco NX-OS does not support the SNMPv3 noAuthNoPriv security level.
- Cisco Nexus 9000 Series switches and the Cisco Nexus 3164Q, 31128PQ, 3232C, and 3264Q switches support the configuration of the SNMP local engine ID.
- For a nondisruptive downgrade path to an earlier release, if a local engine ID has been configured, then you must unconfigure the local engine ID, and then reconfigure the SNMP users and the community strings.
- Special characters @ and % are not allowed in the SNMP community string.
- The default SNMP PDU value is 1500 bytes. The SNMP agent drops any response PDU that is greater than 1500 bytes, causing the SNMP request to fail. To receive MIB data values larger than 1500 bytes, use the **snmp-server packetsize** <byte-count> command to reconfigure the packet size. The valid byte-count range is from 484 to 17382. When a GETBULK response exceeds the packet size, the data can get truncated.
- You must use either the CLI or SNMP to configure a feature on your switch. Do not configure a feature using both interfaces to the switch.
- Using `cefcFanTrayOperStatus snmpwalk` on an individual fan OID tree where the fan is not populated in chassis, can return a response for next OID entry in the tree. To prevent this behavior, use the `-CI` option in `snmpwalk`.
The behavior is not seen when polling parent OID, or when using `getmany`.
- Cisco Nexus 9000 series switches support upto 10000 flash files for `snmpwalk` request.

- There must be at least one running BGP instance to have full, proper functional behavior of SNMP traps. Configure a BGP routing instance before configuring any snmp-server traps related commands.
- Beginning with Release 10.1(1), AES-128 is the recommended encryption algorithm, as it is a strong encryption algorithm. However, DES encryption is also supported.

Downgrade: In-Service System Downgrade (ISSD) with **install all** command is aborted if users with DES privacy protocol are present in the SNMP database. Users need to be reconfigured (using the default AES-128) or deleted. In case of a cold reboot, the SNMP users with DES are deleted.

Default Settings for SNMP

The following table lists the default settings for SNMP parameters.

Parameters	Default
License notifications	Enabled

Configuring SNMP



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.



Note From Cisco NX-OS release 9.3(7), HMAC-SHA-256 authentication protocol is used for SNMPv3.

Configuring SNMP Users

You can configure a user for SNMP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server user <i>name</i> [auth { md5 sha sha-256 } <i>passphrase</i> [auto] [priv [aes-128] <i>passphrase</i>] [engineID <i>id</i>] [localizedkey] [localizedV2key]] Example:	Configures an SNMP user with authentication and privacy parameters. The passphrase can be any case-sensitive, alphanumeric string up to 64 characters. If you use the localizedkey keyword, the passphrase can be any

	Command or Action	Purpose
	<pre>switch(config)# snmp-server user Admin pwd_type 6 auth sha abcd1234 priv abcdefgh</pre>	<p>case-sensitive, alphanumeric string up to 130 characters.</p> <p>localizedkey - If you use the localizedkey keyword, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters. Instead of plain-text password, hashed password (copied either from the show running config command or generated offline using snmpv3 based open source hash generator tool, see Generating Hashed Password Offline, on page 9) can be configured using the localizedkey keyword.</p> <p>Note When using a localized key, add <code>0x</code> before the hash value, for example, <code>0x84a716329158a97ac9f22780629bc26c</code>.</p> <p>localizedV2key - If the localizedV2key is used, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters, without <code>0x</code> at the beginning. Collect the localizedv2key using show run command, as this is an encrypted data and cannot be generated offline.</p> <p>The engineID format is a 12-digit, colon-separated decimal number.</p> <p>Note Beginning with Release 10.1(1), AES-128 is the default privacy protocol for SNMPv3.</p>
Step 3	<p>(Optional) show snmp user</p> <p>Example:</p> <pre>switch(config)# show snmp user</pre>	Displays information about one or more SNMP users.
Step 4	<p>(Optional) copy running-config startup-config</p> <p>Example:</p> <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Generating Hashed Password Offline

Perform the following steps to generate hashed password offline, using snmpv3-based open source hash generator tool:



Note The IDs mentioned in this procedure are only sample IDs, the purpose of which is only to explain the procedure better.

1. Get the SNMP engineID from the switch.

```
switch# show snmp engineID
```

Sample output:

```
Local SNMP engineID: [Hex] 8000000903D4C93CEA31CC
[Dec] 128:000:000:009:003:212:201:060:234:049:204
```

2. Use an SNMPv3 based open source hash generator to generate offline hashed password.

```
Linux$ snmpv3-hashgen --auth Hello123 --engine 8000000903D4C93CEA31CC --user1 --mode priv
--hash md5
```

Sample output:

```
User: user1
Auth: Hello123 / 84a716329158a97ac9f22780629bc26c
Priv: Hello123 / 84a716329158a97ac9f22780629bc26c
Engine: 8000000903D4C93CEA31CC
ESXi USM String: u1/84a716329158a97ac9f22780629bc26c/84a716329158a97ac9f22780629bc26c/priv
```

3. Use the auth and priv values to configure the password on the switch.

```
snmp-server user user1 auth md5 0x84a716329158a97ac9f22780629bc26c priv des
0x84a716329158a97ac9f22780629bc26c localizedkey
```

Enforcing SNMP Message Encryption

You can configure SNMP to require authentication or encryption for incoming requests. By default, the SNMP agent accepts SNMPv3 messages without authentication and encryption. When you enforce privacy, Cisco NX-OS responds with an authorization error for any SNMPv3 PDU request using a security level parameter of either **noAuthNoPriv** or **authNoPriv**.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server user <i>name</i> enforcePriv Example: switch(config)# snmp-server user Admin enforcePriv	Enforces SNMP message encryption for this user.
Step 3	snmp-server globalEnforcePriv Example:	Enforces SNMP message encryption for all users.

	Command or Action	Purpose
	switch(config)# snmp-server globalEnforcePriv	
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Assigning SNMPv3 Users to Multiple Roles

After you configure an SNMP user, you can assign multiple roles for the user.



Note Only users belonging to a network-admin role can assign roles to other users.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server user <i>name group</i> Example: switch(config)# snmp-server user Admin superuser	Associates this SNMP user with the configured user role.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Creating SNMP Communities

You can create SNMP communities for SNMPv1 or SNMPv2c.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	snmp-server community <i>name</i> { group <i>group</i> ro rw } Example: <pre>switch(config)# snmp-server community public ro</pre>	Creates an SNMP community string.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Filtering SNMP Requests

You can assign an access control list (ACL) to an SNMPv2 community to filter SNMP requests. If the assigned ACL allows the incoming request packet, SNMP processes the request. If the ACL denies the request, SNMP drops the request and sends a system message.

Create the ACL with the following parameters:

- Source IP address
- Destination IP address
- Source port
- Destination port
- Protocol (UDP or TCP)

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server community <i>name</i> [use-ipv4acl <i>acl-name</i>] Example: <pre>switch(config)# snmp-server community public use-ipv4acl myacl</pre>	Assigns an IPv4 ACL to an SNMPv2 community to filter SNMP requests.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring SNMP Notification Receivers

You can configure Cisco NX-OS to generate SNMP notifications to multiple host receivers.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server host ip-address traps version 1 community [udp_port number] Example: <pre>switch(config)# snmp-server host 192.0.2.1 traps version 1 public</pre>	Configures a host receiver for SNMPv1 traps. The <i>ip-address</i> can be an IPv4 or IPv6 address. The <i>community</i> can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.
Step 3	snmp-server host ip-address {traps informs} version 2c community [udp_port number] Example: <pre>switch(config)# snmp-server host 192.0.2.1 informs version 2c public</pre>	Configures a host receiver for SNMPv2c traps or informs. The <i>ip-address</i> can be an IPv4 or IPv6 address. The <i>community</i> can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.
Step 4	snmp-server host ip-address {traps informs} version 3 {auth noauth priv} username [udp_port number] Example: <pre>switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS</pre>	Configures a host receiver for SNMPv3 traps or informs. The <i>ip-address</i> can be an IPv4 or IPv6 address. The <i>username</i> can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. Note The SNMP manager must know the user credentials (authKey/PrivKey) based on the SNMP engine ID of the Cisco NX-OS device to authenticate and decrypt the SNMPv3 messages.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring a Source Interface for SNMP Notifications

You can configure SNMP to use the IP address of an interface as the source IP address for notifications. When a notification is generated, its source IP address is based on the IP address of this configured interface.

You can configure a source interface as follows:

- All notifications sent to all SNMP notification receivers.
- All notifications sent to a specific SNMP notification receiver. This configuration overrides the global source interface configuration.



Note Configuring the source interface IP address for outgoing trap packets does not guarantee that the device will use the same interface to send the trap. The source interface IP address defines the source address inside of the SNMP trap, and the connection is opened with the address of the egress interface as source.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server host ip-address source-interface if-type if-number traps version 2c name Example: <pre>snmp-server host 192.0.2.1 source-interface ethernet 2/1 traps version 2c public</pre>	(Optional) Send Traps messages to this host. The traps version is the SNMP version to use for notification messages. 2c indicates that SNMPv2c is to be used.
Step 3	snmp-server host ip-address source-interface if-type if-number use-vrf vrf-name Example: <pre>snmp-server host 192.0.2.1 source-interface ethernet 2/1 use-vrf default</pre>	Configures SNMP to use the selected VRF to communicate with the host receiver. The ip-address can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 32 characters. Note This command does not remove the host configuration.
Step 4	snmp-server host ip-address source-interface if-type if-number [udp_port number] Example: <pre>switch(config)# snmp-server host 192.0.2.1 source-interface ethernet 2/1</pre>	Configures a host receiver for SNMPv2c traps or informs. The ip-address can be an IPv4 or IPv6 address. Use ? to determine the supported interface types. The UDP port number range is from 0 to 65535. This configuration overrides the global source interface configuration.
Step 5	snmp-server source-interface {traps informs} if-type if-number Example: <pre>switch(config)# snmp-server source-interface traps ethernet 2/1</pre>	Configures a source interface for sending out SNMPv2c traps or informs. Use ? to determine the supported interface types.

	Command or Action	Purpose
Step 6	show snmp source-interface Example: <pre>switch(config)# show snmp source-interface</pre>	Displays information about configured source interfaces.

Configuring the Notification Target User

You must configure a notification target user on the device to send SNMPv3 inform notifications to a notification host receiver.

Cisco NX-OS uses the credentials of the notification target user to encrypt the SNMPv3 inform notification messages to the configured notification host receiver.



Note For authenticating and decrypting the received inform PDU, the notification host receiver should have the same user credentials as configured in Cisco NX-OS to authenticate and decrypt the informs.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server user name [auth {md5 sha sha-256} passphrase [auto] [priv passphrase] [engineID id] Example: <pre>switch(config)# snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID 00:00:00:63:00:01:00:10:20:15:10:03</pre>	Configures the notification target user with the specified engine ID for the notification host receiver. The engine ID format is a 12-digit colon-separated decimal number. Note Beginning with Release 10.1(1), AES-128 is the default privacy protocol for SNMPv3.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring SNMP Notification Receivers with VRFs

SNMP adds entries into the cExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MIB when you configure the VRF reachability and filtering options for an SNMP notification receiver.



Note You must configure the host before configuring the VRF reachability or filtering options.

You can configure Cisco NX-OS to use a configured VRF to reach the host receiver or to filter notifications based on the VRF in which the notification occurred.

Procedure

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	<p>[no] snmp-server host ip-address use-vrf vrf-name [udp_port number]</p> <p>Example:</p> <pre>switch(config)# snmp-server host 192.0.2.1 use-vrf Blue</pre>	<p>Configures SNMP to use the selected VRF to communicate with the host receiver. The <i>ip-address</i> can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into the ExtSnmptargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.</p> <p>The no form of this command removes the VRF reachability information for the configured host and removes the entry from the ExtSnmptargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.</p> <p>Note This command does not remove the host configuration.</p>
Step 3	<p>[no] snmp-server host ip-address filter-vrf vrf-name [udp_port number]</p> <p>Example:</p> <pre>switch(config)# snmp-server host 192.0.2.1 filter-vrf Red</pre>	<p>Filters notifications to the notification host receiver based on the configured VRF. The <i>ip-address</i> can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.</p> <p>This command adds an entry into the ExtSnmptargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.</p> <p>The no form of this command removes the VRF filter information for the configured host and removes the entry from the ExtSnmptargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.</p> <p>Note This command does not remove the host configuration.</p>

	Command or Action	Purpose
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring SNMP to Send Traps Using an Inband Port

You can configure SNMP to send traps using an inband port. To do so, you must configure the source interface (at the global or host level) and the VRF used to send the traps.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server source-interface traps if-type if-number Example: <pre>switch(config)# snmp-server source-interface traps ethernet 1/2</pre>	Globally configures a source interface for sending out SNMP traps. Use ? to determine the supported interface types. You can configure the source interface at the global level or at a host level. When the source interface is configured globally, any new host configuration uses the global configuration to send the traps. Note To configure a source interface at the host level, use the snmp-server host ip-address source-interface if-type if-number command.
Step 3	(Optional) show snmp source-interface Example: <pre>switch(config)# show snmp source-interface</pre>	Displays information about configured source interfaces.
Step 4	snmp-server host ip-address use-vrf vrf-name [udp_port number] Example: <pre>switch(config)# snmp-server host 171.71.48.164 use-vrf default</pre>	Configures SNMP to use the selected VRF to communicate with the host receiver. The <i>ip-address</i> can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into the ExtSnmptargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB.

	Command or Action	Purpose
		Note By default, SNMP sends the traps using the management VRF. If you do not want to use the management VRF, you must use this command to specify the desired VRF.
Step 5	(Optional) show snmp host Example: switch(config)# show snmp host	Displays information about configured SNMP hosts.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling SNMP Notifications

You can enable or disable notifications. If you do not specify a notification name, Cisco NX-OS enables all notifications except BGP, EIGRP, and OSPF notifications.



Note The **snmp-server enable traps** command enables both traps and informs, depending on the configured notification host receivers.

The following table lists the commands that enable the notifications for Cisco NX-OS MIBs.

Table 2: Enabling SNMP Notifications

MIB	Related Commands
All notifications (except BGP, EIGRP, and OSPF)	snmp-server enable traps
CISCO-AAA-SERVER-MIB	snmp-server enable traps aaa snmp-server enable traps aaa server-state-change
CISCO-BGP4-MIB	snmp-server enable traps bgp
CISCO-CALLHOME-MIB	snmp-server enable traps callhome snmp-server enable traps callhome event-notify snmp-server enable traps callhome smtp-send-fail
CISCO-CONFIG-MAN-MIB	snmp-server enable traps config snmp-server enable traps config ccmCLIRunningConfigChanged

MIB	Related Commands
CISCO-EIGRP-MIB	<code>snmp-server enable traps eigrp [tag]</code>
CISCO-ERR-DISABLE-MIB	<code>snmp-server enable traps link cerrDisableInterfaceEventRev1</code>
ENTITY-MIB, CISCO-ENTITY-SENSOR-MIB	<code>snmp-server enable traps entity</code> <code>snmp-server enable traps entity entity_fan_status_change</code> <code>snmp-server enable traps entity entity_mib_change</code> <code>snmp-server enable traps entity entity_module_inserted</code> <code>snmp-server enable traps entity entity_module_removed</code> <code>snmp-server enable traps entity entity_module_status_change</code> <code>snmp-server enable traps entity entity_power_out_change</code> <code>snmp-server enable traps entity entity_power_status_change</code> <code>snmp-server enable traps entity entity_unrecognised_module</code>
CISCO-FEATURE-CONTROL-MIB	<code>snmp-server enable traps feature-control</code> <code>snmp-server enable traps feature-control FeatureOpStatusChange</code>
CISCO-HSRP-MIB	<code>snmp-server enable traps hsrp</code> <code>snmp-server enable traps hsrp state-change</code>
CISCO-LICENSE-MGR-MIB	<code>snmp-server enable traps license</code> <code>snmp-server enable traps license notify-license-expiry</code> <code>snmp-server enable traps license notify-license-expiry-warning</code> <code>snmp-server enable traps license notify-licensefile-missing</code> <code>snmp-server enable traps license notify-no-license-for-feature</code>

MIB	Related Commands
IF-MIB	snmp-server enable traps link snmp-server enable traps link IETF-extended-linkDown snmp-server enable traps link IETF-extended-linkUp snmp-server enable traps link cisco-extended-linkDown snmp-server enable traps link cisco-extended-linkUp snmp-server enable traps link linkDown snmp-server enable traps link Up
OSPF-MIB, OSPF-TRAP-MIB	snmp-server enable traps ospf [tag] snmp-server enable traps ospf lsa snmp-server enable traps ospf rate-limit rate
CISCO-RF-MIB	snmp-server enable traps rf snmp-server enable traps rf redundancy_framework
CISCO-RMON-MIB	snmp-server enable traps rmon snmp-server enable traps rmon fallingAlarm snmp-server enable traps rmon hcFallingAlarm snmp-server enable traps rmon hcRisingAlarm snmp-server enable traps rmon risingAlarm
SNMPv2-MIB	snmp-server enable traps snmp snmp-server enable traps snmp authentication
CISCO-MAC-NOTIFICATION-MIB	snmp-server enable trap link cmn-mac-move-notification
CISCO-PORT-STORM-CONTROL-MIB	storm-control action trap
CISCO-STP-EXTENSIONS-MIB	snmp-server enable traps stpx stpxMstInconsistencyUpdate
CISCO-STP-BRIDGE-MIB	snmp-server enable traps bridge snmp-server enable traps bridge newroot snmp-server enable traps bridge topologychange

MIB	Related Commands
CISCO-STPX-MIB	snmp-server enable traps stpx snmp-server enable traps stpx inconsistency snmp-server enable traps stpx loop-inconsistency snmp-server enable traps stpx root-inconsistency
CISCO-SYSTEM-EXT-MIB	snmp-server enable traps sysmgr snmp-server enable traps sysmgr cseFailSwCoreNotifyExtended
UPGRADE-MIB	snmp-server enable traps upgrade snmp-server enable traps upgrade UpgradeJobStatusNotify snmp-server enable traps upgrade UpgradeOpNotifyOnCompletion
VTP-MIB	snmp-server enable traps vtp snmp-server enable traps vtp notif snmp-server enable traps vtp vlancreate snmp-server enable traps vtp vlandelete

Use the following commands in the configuration mode shown to enable the specified notification:

Command	Purpose
snmp-server enable traps Example: <pre>switch(config)# snmp-server enable traps</pre>	Enables all SNMP notifications.
snmp-server enable traps aaa [server-state-change] Example: <pre>switch(config)# snmp-server enable traps aaa</pre>	Enables the AAA SNMP notifications. Optionally, enables the following specific notifications: <ul style="list-style-type: none"> • server-state-change—Enables AAA server state-change notifications.
snmp-server enable traps bgp Example: <pre>switch(config)# snmp-server enable traps bgp</pre>	Enables Border Gateway Protocol (BGP) SNMP notifications.
snmp-server enable traps bridge [newroot] [topologychange] Example: <pre>switch(config)# snmp-server enable traps bridge</pre>	Enables STP bridge SNMP notifications. Optionally, enables the following specific notifications: <ul style="list-style-type: none"> • newroot—Enables STP new root bridge notifications. • topologychange—Enables STP bridge topology-change notifications.

Command	Purpose
<p>snmp-server enable traps callhome [event-notify] [smtp-send-fail]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps callhome</pre>	<p>Enables Call Home notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • event-notify—Enables Call Home external event notifications. • smtp-send-fail—Enables Simple Mail Transfer Protocol (SMTP) message send fail notifications.
<p>snmp-server enable traps config [ccmCLIRunningConfigChanged]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps config</pre>	<p>Enables SNMP notifications for configuration changes.</p> <ul style="list-style-type: none"> • ccmCLIRunningConfigChanged—Enables SNMP notifications for configuration changes in the running or startup configuration.
<p>snmp-server enable traps eigrp [tag]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps eigrp</pre>	<p>Enables CISCO-EIGRP-MIB SNMP notifications.</p>
<p>snmp-server enable traps entity [entity_fan_status_change] [entity_mib_change] [entity_module_inserted] [entity_module_removed] [entity_module_status_change] [entity_power_out_change] [entity_power_status_change] [entity_unrecognised_module]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps entity</pre>	<p>Enables ENTITY-MIB SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • entity_fan_status_change—Enables entity fan status-change notifications. • entity_mib_change—Enables entity MIB change notifications. • entity_module_inserted—Enables entity module inserted notifications. • entity_module_removed—Enables entity module removed notifications. • entity_module_status_change—Enables entity module status-change notifications. • entity_power_out_change—Enables entity power-out change notifications. • entity_power_status_change—Enables entity power status-change notifications. • entity_unrecognised_module—Enables entity unrecognized module notifications.

Command	Purpose
<p>snmp-server enable traps feature-control [FeatureOpStatusChange]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps feature-control</pre>	<p>Enables feature-control SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • FeatureOpStatusChange—Enables feature operation status-change notifications.
<p>snmp-server enable traps hsrp state-change</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps hsrp</pre>	<p>Enables CISCO-HSRP-MIB SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • state-change—Enables HSRP state-change notifications.
<p>snmp-server enable traps license [notify-license-expiry] [notify-license-expiry-warning] [notify-licensefile-missing] [notify-no-license-for-feature]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps license</pre>	<p>Enables ENTITY-MIB SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • notify-license-expiry—Enables license expiry notifications. • notify-license-expiry-warning—Enables license expiry warning notifications. • notify-licensefile-missing—Enables license file-missing notifications. • notify-no-license-for-feature—Enables no-license-installed-for-feature notifications.

Command	Purpose
<p>snmp-server enable traps link [cieLinkDown] [cieLinkUp] [cmn-mac-move-notification] [IETF-extended-linkDown] [IETF-extended-linkUp] [cisco-extended-linkDown] [cisco-extended-linkUp][linkDown] [linkUp]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps link</pre>	<p>Enables IF-MIB link notifications. Optionally, enable the following specific notifications:</p> <ul style="list-style-type: none"> • IETF-extended-linkDown—Enables Cisco extended link state down notifications. • IETF-extended-linkUp—Enables Cisco extended link state up notifications. • cmn-mac-move-notification—Enables MAC address move notifications. • cisco-extended-linkDown—Enables Internet Engineering Task Force (IETF) extended link state down notifications. • cisco-extended-linkUp—Enables Internet Engineering Task Force (IETF) extended link state up notifications. • linkDown—Enables IETF link state down notifications. • linkUp—Enables IETF link state up notifications.
<p>snmp-server enable traps ospf [tag] [lsa]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps ospf</pre>	<p>Enables Open Shortest Path First (OSPF) notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • lsa—Enables OSPF link state advertisement (LSA) notifications.
<p>snmp-server enable traps rf [redundancy-framework]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps rf</pre>	<p>Enables redundancy framework (RF) SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • redundancy-framework—Enables RF supervisor switchover MIB notifications.

Command	Purpose
<p>snmp-server enable traps rmon [fallingAlarm] [hcFallingAlarm] [hcRisingAlarm] [risingAlarm]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps rmon</pre>	<p>Enables remote monitoring (RMON) SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • fallingAlarm—Enables RMON falling alarm notifications. • hcFallingAlarm—Enables RMON high-capacity falling alarm notifications. • hcRisingAlarm—Enables RMON high-capacity rising alarm notifications. • risingAlarm—Enables RMON rising alarm notifications.
<p>snmp-server enable traps snmp [authentication]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps snmp</pre>	<p>Enables general SNMP notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • authentication—Enables SNMP authentication notifications.
<p>snmp-server enable traps stpx [inconsistency] [loop-inconsistency] [root-inconsistency]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps stpx</pre>	<p>Enables SNMP STPX notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • inconsistency—Enables SNMP STPX MIB inconsistency update notifications. • loop-inconsistency—Enables SNMP STPX MIB loop-inconsistency update notifications. • root-inconsistency—Enables SNMP STPX MIB root-inconsistency update notifications.
<p>snmp-server enable traps syslog [message-generated]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps syslog</pre>	<p>Sends syslog messages as traps to the defined SNMP host. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • message-generated—Enables software log message generated notifications.
<p>snmp-server enable traps sysmgr [cseFailSwCoreNotifyExtended]</p> <p>Example:</p> <pre>switch(config)# snmp-server enable traps sysmgr</pre>	<p>Enables software change notifications. Optionally, enables the following specific notifications:</p> <ul style="list-style-type: none"> • cseFailSwCoreNotifyExtended—Enables software core notifications.

Command	Purpose
snmp-server enable traps upgrade [UpgradeJobStatusNotify] [UpgradeOpNotifyOnCompletion] Example: <pre>switch(config)# snmp-server enable traps upgrade</pre>	Enables upgrade notifications. Optionally, enables the following specific notifications: <ul style="list-style-type: none"> • UpgradeJobStatusNotify—Enables upgrade job status notifications. • UpgradeOpNotifyOnCompletion—Enables upgrade global status notifications.
snmp-server enable traps vtp [notifs] [vlancreate] [vlandelete] Example: <pre>switch(config)# snmp-server enable traps vtp</pre>	Enables VTP notifications. Optionally, enables the following specific notifications: <ul style="list-style-type: none"> • notifs—Enables VTP notifications. • vlancreate—Enables VLAN creation notifications. • vlandelete—Enables VLAN deletion notifications.
storm-control action traps Example: <pre>switch(config-if)# storm-control action traps</pre>	Enables traffic storm control notifications when the traffic storm control limit is reached.

Disabling Link Notifications on an Interface

You can disable linkUp and linkDown notifications on an individual interface. You can use this limit notifications on flapping interface (an interface that transitions between up and down repeatedly).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface type slot/port Example: <pre>switch(config)# interface ethernet 2/2</pre>	Disables SNMP link-state traps for the interface. This command is enabled by default.
Step 3	no snmp trap link-status Example: <pre>switch(config-if)# no snmp trap link-status</pre>	Disables SNMP link-state traps for the interface. This command is enabled by default.

	Command or Action	Purpose
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Displaying SNMP ifIndex for an Interface

The SNMP ifIndex is used across multiple SNMP MIBs to link related interface information.

Procedure

	Command or Action	Purpose
Step 1	show interface snmp-ifindex Example: <pre>switch# show interface snmp-ifindex grep -i Eth12/1 Eth12/1 441974784 (0x1a580000)</pre>	Displays the persistent SNMP ifIndex value from the IF-MIB for all interfaces. Optionally, use the keyword and the grep keyword to search for a particular interface in the output.

Enabling a One-Time Authentication for SNMP over TCP

You can enable a one-time authentication for SNMP over a TCP session.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server tcp-session [auth] Example: <pre>switch(config)# snmp-server tcp-session</pre>	Enables a one-time authentication for SNMP over a TCP session. The default is disabled.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Assigning SNMP Device Contact and Location Information

You can assign the device contact information, which is limited to 32 characters (without spaces) and the device location.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server contact <i>name</i> Example: switch(config)# snmp-server contact Admin	Configures sysContact, which is the SNMP contact name.
Step 3	snmp-server location <i>name</i> Example: switch(config)# snmp-server location Lab-7	Configures sysLocation, which is the SNMP location.
Step 4	(Optional) show snmp Example: switch(config)# show snmp	Displays information about one or more destination profiles.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the Context to Network Entity Mapping

You can configure an SNMP context to map to a logical network entity, such as a protocol instance or VRF.

Before you begin

Determine the logical network entity instance. For more information on VRFs and protocol instances, see the [Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide](#) or the [Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] snmp-server context <i>context-name</i> [instance <i>instance-name</i>] [vrf <i>vrf-name</i>] [topology <i>topology-name</i>]	Maps an SNMP context to a protocol instance, VRF, or topology. The names can be any alphanumeric string up to 32 characters.

	Command or Action	Purpose
	Example: <pre>switch(config)# snmp-server context public1 vrf red</pre>	<p>The no option deletes the mapping between an SNMP context and a protocol instance, VRF, or topology.</p> <p>Note Do not enter an instance, VRF, or topology to delete a context mapping. If you use the instance, VRF, or topology keywords, you configure a mapping between the context and a zero-length string.</p>
Step 3	(Optional) snmp-server mib community-map <i>community-name context context-name</i> Example: <pre>switch(config)# snmp-server mib community-map public context public1</pre>	Maps an SNMPv2c community to an SNMP context. The names can be any alphanumeric string up to 32 characters.
Step 4	(Optional) show snmp context Example: <pre>switch(config)# show snmp context</pre>	Displays information about one or more SNMP contexts.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Disabling SNMP

You can disable SNMP on the device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	no snmp-server protocol enable Example: <pre>switch(config)# no snmp-server protocol enable</pre>	<p>Disables SNMP. SNMP is enabled by default.</p> <p>Note You cannot disable SNMPv1 without disabling SNMPv2. If you want to disable SNMPv1, then configure only SNMPv3, or disable SNMP entirely.</p>

Managing the SNMP Server Counter Cache Update Timer

You can modify how long, in seconds Cisco NX-OS holds the cache port state.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server counter cache timeout <i>seconds</i> Example: <pre>switch(config)# snmp-server counter cache timeout 1200</pre>	Defines how long in seconds, the port states are held in the local cache. The counter cache is enabled by default, and the default cache timeout value is 10 seconds. When disabled, the default cache timeout value is 50 seconds. The range is 1-3600. Note For end of row (EoR) switching - The range is from 10 to 3600.
Step 3	(Optional) show running-config snmp all i cac Example: <pre>switch(config)# copy running-config snmp all i cac</pre>	Displays the configured SNMP-server counter cache update timeout value.
Step 4	no snmp-server counter cache enable Example: <pre>switch(config)# no snmp-server counter cache enable</pre>	Disables the counter cache update. Note When the counter cache update is disabled, the value set in the timeout parameter determines length of time the port states are held the counter cache.

Modifying the AAA Synchronization Time

You can modify how long Cisco NX-OS holds the synchronized user configuration.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	snmp-server aaa-user cache-timeout seconds Example: <pre>switch(config)# snmp-server aaa-user cache-timeout 1200</pre>	Configures how long the AAA synchronized user configuration stays in the local cache. The range is from 1 to 86400 seconds. The default is 3600.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the SNMP Local Engine ID

Beginning with Cisco NX-OS Release 7.0(3)I6(1), you can configure the engine ID on a local device.



Note After you configure the SNMP local engine ID, you must reconfigure all SNMP users, any host configured with the V3 users, and the community strings. Beginning with Cisco NX-OS Release 7.0(3)I7(1), you need to reconfigure only the SNMP users and community strings.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server engineID local engineid-string Example: <pre>switch(config)# snmp-server engineID local AA:BB:CC:1A:2C:10</pre>	<p>Changes the SNMP engine ID of the local device.</p> <p>The local engine ID should be configured as a list of colon-specified hexadecimal octets, where there are even number of hexadecimal characters that range from 10 to 64 and every two hexadecimal characters are separated by a colon. For example, 80:00:02:b8:04:61:62:63.</p>
Step 3	show snmp engineID Example: <pre>switch(config)# show snmp engineID</pre>	Displays the identification of the configured SNMP engine.
Step 4	[no] snmp-server engineID local engineid-string Example:	Disables the local engine ID and the default auto-generated engine ID is configured.

	Command or Action	Purpose
	<pre>switch(config)# no snmp-server engineID local AA:BB:CC:1A:2C:10</pre>	
Step 5	<p>Required: copy running-config startup-config</p> <p>Example:</p> <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying SNMP Configuration

To display SNMP configuration information, perform one of the following tasks:

Command	Purpose
show interface snmp-ifindex	Displays the SNMP ifIndex value for all interfaces (from IF-MIB).
show running-config snmp [all]	<p>Displays the SNMP running configuration.</p> <p>SNMP users brought into 10.1(1), from releases prior to 10.1(1), are displayed with the configured privacy protocol, AES-128 or DES. New users (Release 10.1(1) and later) are by default configured with AES-128 protocol.</p> <p>Beginning with 9.3(8) release, SNMPv3 users under show run will be represented in SALT format instead of hash.</p>
show snmp	Displays the SNMP status.

Command	Purpose
show snmp community	Displays the SNMP community strings. Note If the name of the SNMP context in the snmp-server mib community-map command is more than 11 characters, the output of the show snmp community command is displayed in a vertical format instead of a tabular format.
show snmp context	Displays the SNMP context mapping.
show snmp engineID	Displays the SNMP engineID.
show snmp group	Displays SNMP roles.
show snmp host	Displays information about configured SNMP hosts.
show snmp session	Displays SNMP sessions.
show snmp source-interface	Displays information about configured source interfaces.
show snmp trap	Displays the SNMP notifications enabled or disabled.
show snmp user	Displays SNMPv3 users.

Configuration Examples for SNMP

This example shows how to configure Cisco NX-OS to send the Cisco linkUp or Down notifications to one notification host receiver using the Blue VRF and defines two SNMP users, Admin and NMS:

```

configure terminal
snmp-server contact Admin@company.com
snmp-server user Admin auth sha abcd1234 priv abcdefgh
snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID
00:00:00:63:00:01:00:22:32:15:10:03
snmp-server host 192.0.2.1 informs version 3 auth NMS
snmp-server host 192.0.2.1 use-vrf Blue
snmp-server enable traps link cisco

```

This example shows how to configure SNMP to send traps using an inband port configured at the host level:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server host 171.71.48.164 version 2c public
switch(config)# snmp-server host 171.71.48.164 source-interface ethernet 1/2
switch(config)# show snmp host
-----
Host Port Version Level Type SecName
-----
171.71.48.164 162 v2c noauth trap public
Source interface: Ethernet 1/2
-----
switch(config)# snmp-server host 171.71.48.164 use-vrf default
switch(config)# show snmp host
-----
Host Port Version Level Type SecName
-----
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-----
```

This example shows how to configure SNMP to send traps using a globally configured inband port:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server source-interface traps ethernet 1/2
switch(config)# show snmp source-interface
-----
Notification source-interface
-----
trap Ethernet1/2
inform -
-----
switch(config)# snmp-server host 171.71.48.164 use_vrf default
switch(config)# show snmp host
-----
Host Port Version Level Type SecName
-----
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-----
```

This example shows how to map VRF red to the SNMPv2c public community string:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# vrf context red
switch(config-vrf)# exit
switch(config)# snmp-server context public1 vrf red
switch(config)# snmp-server mib community-map public context public1
```

This example shows how to map OSPF instance Enterprise to the same SNMPv2c public community string:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature ospf
switch(config)# router ospf Enterprise
switch(config-router)# exit
switch(config)# snmp-server context public1 instance Enterprise
```

```
switch(config)# snmp-server mib community-map public context public1
```

Additional References

Related Documents

Related Topic	Document Title
IP ACLs and AAA	<i>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</i>
MIBs	<i>Cisco Nexus 7000 Series and 9000 Series NX-OS MIB Quick Reference</i>

RFCs

RFC	Title
RFC 3414	<i>User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3)</i>
RFC 3415	<i>View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)</i>

MIBs

MIBs	MIBs Link
MIBs related to SNMP	To locate and download supported MIBs, go to the following link: https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

