Implementing SNMP

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.

This module describes the new and revised tasks you need to implement SNMP on your Cisco IOS XR network.

For detailed conceptual information about SNMP on the Cisco IOS XR software and complete descriptions of the SNMP commands listed in this module, see Related Documents, on page 25. For information on specific MIBs, refer to Cisco CRS and Cisco XR 12000 Series Router MIB Overview. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in Cisco IOS XR Commands Master List for the Cisco CRS Router.

Table 1: Feature History for Implementing SNMP on Cisco IOS XR Software

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 2.0</td>
<td>This feature was introduced.</td>
</tr>
<tr>
<td>Release 3.6.0</td>
<td>Support was added for encrypted community strings. Support was added for setting IPv4 precedence and DSCP values. The clear snmp counters command was added.</td>
</tr>
<tr>
<td>Release 3.8.0</td>
<td>Support was added for SNMP contexts.</td>
</tr>
<tr>
<td>Release 3.9.0</td>
<td>Support was added for packet loss monitoring. Support was added for 3DES and AES encryption. The ability to preserve ENTITY-MIB and CISCO-CLASS-BASED-QOS-MIB data was added.</td>
</tr>
<tr>
<td>Release 4.2.0</td>
<td>Support was added for SNMP over IPv6.</td>
</tr>
</tbody>
</table>

This module contains the following topics:

- Prerequisites for Implementing SNMP, page 2
Prerequisites for Implementing SNMP

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Restrictions for SNMP Use on Cisco IOS XR Software

SNMP outputs are only 32-bits wide and therefore cannot display any information greater than \(2^{32}\). \(2^{32}\) is equal to 4.29 Gigabits. Note that a 10 Gigabit interface is greater than this and so if you are trying to display speed information regarding the interface, you might see concatenated results.

Information About Implementing SNMP

To implement SNMP, you need to understand the concepts described in this section.

SNMP Functional Overview

The SNMP framework consists of three parts:

- SNMP manager
- SNMP agent
- Management Information Base (MIB)

SNMP Manager

The SNMP manager is the system used to control and monitor the activities of network hosts using SNMP. The most common managing system is called a network management system (NMS). The term NMS can be applied to either a dedicated device used for network management, or the applications used on such a device. A variety of network management applications are available for use with SNMP. These features range from simple command-line applications to feature-rich graphical user interfaces (such as the CiscoWorks 2000 line of products).
SNMP Agent

The SNMP agent is the software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The agent and MIB reside on the router. To enable the SNMP agent, you must define the relationship between the manager and the agent.

MIB

The Management Information Base (MIB) is a virtual information storage area for network management information, which consists of collections of managed objects. Within the MIB there are collections of related objects, defined in MIB modules. MIB modules are written in the SNMP MIB module language, as defined in STD 58, RFC 2578, RFC 2579, and RFC 2580. Note that individual MIB modules are also referred to as MIBs; for example, the Interfaces Group MIB (IF-MIB) is a MIB module within the MIB on your system.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change through Get or Set operations. A manager can get a value from an agent or store a value into that agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to manager requests to get or set data.

Figure 1: Communication Between an SNMP Agent and Manager, on page 3 illustrates the communications relationship between the SNMP manager and agent. A manager can send the agent requests to get and set MIB values. The agent can respond to these requests. Independent of this interaction, the agent can send unsolicited notifications (traps) to the manager to notify the manager of network conditions.

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. On Cisco IOS XR software, unsolicited (asynchronous) notifications can be generated only as traps. Traps are messages alerting the SNMP manager to a condition on the network. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Note

Inform requests (inform operations) are not supported in Cisco IOS XR software.

Traps are less reliable than informs because the receiver does not send any acknowledgment when it receives a trap. The sender 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 manager does not receive an inform request, it does not send a response. If the sender never receives a response, the inform request can be sent again. Thus, informs are more likely to reach their intended destination.
However, traps are often preferred because informs consume more resources in the router and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once, and an inform may be retried several times. The retries increase traffic and contribute to a higher overhead on the network. Thus, traps and inform requests provide a trade-off between reliability and resources.

In this illustration, the agent router sends a trap to the SNMP manager. Although the manager receives the trap, it does not send any acknowledgment to the agent. The agent has no way of knowing that the trap reached its destination.

Figure 2: Trap Received by the SNMP Manager

In this illustration, the agent sends a trap to the manager, but the trap does not reach the manager. Because the agent has no way of knowing that the trap did not reach its destination, the trap is not sent again. The manager never receives the trap.

Figure 3: Trap Not Received by the SNMP Manager

SNMP Versions

Cisco IOS XR software supports the following versions of SNMP:

- Simple Network Management Protocol Version 1 (SNMPv1)
- Simple Network Management Protocol Version 2c (SNMPv2c)
- Simple Network Management Protocol Version 3 (SNMPv3)

Both SNMPv1 and SNMPv2c use a community-based form of security. The community of managers able to access the agent MIB is defined by an IP address access control list and password.

SNMPv2c support includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism supports the retrieval of tables and large quantities of
information, minimizing the number of round-trips required. The SNMPv2c improved error handling support includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes now report the error type. Three kinds of exceptions are also reported: no such object exceptions, no such instance exceptions, and end of MIB view exceptions.

SNMPv3 is a security model. A security model is an authentication strategy that is set up for a user and the group 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 will determine which security mechanism is employed when an SNMP packet is handled. See Table 3: SNMP Security Models and Levels, on page 6 for a list of security levels available in SNMPv3. The SNMPv3 feature supports RFCs 3411 to 3418.

You must configure the SNMP agent to use the version of SNMP supported by the management station. An agent can communicate with multiple managers; for this reason, you can configure the Cisco IOS-XR software to support communications with one management station using the SNMPv1 protocol, one using the SNMPv2c protocol, and another using SNMPv3.

**Comparison of SNMPv1, v2c, and v3**

SNMP v1, v2c, and v3 all support the following operations:

- **get-request**—Retrieves a value from a specific variable.
- **get-next-request**—Retrieves the value following the named variable; this operation is often used to retrieve variables from within a table. With this operation, an SNMP manager does not need to know the exact variable name. The SNMP manager searches sequentially to find the needed variable from within the MIB.
- **get-response**—Operation that replies to a get-request, get-next-request, and set-request sent by an NMS.
- **set-request**—Operation that stores a value in a specific variable.
- **trap**—Unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

Table 2: SNMPv1, v2c, and v3 Feature Support, on page 5 identifies other key SNMP features supported by the SNMP v1, v2c, and v3.

<table>
<thead>
<tr>
<th>Feature</th>
<th>SNMP v1</th>
<th>SNMP v2c</th>
<th>SNMP v3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get-Bulk Operation</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Inform Operation</td>
<td>No</td>
<td>Yes (No on the Cisco IOS XR software)</td>
<td>Yes (No on the Cisco IOS XR software)</td>
</tr>
<tr>
<td>64 Bit Counter</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Textual Conventions</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Authentication</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Privacy (Encryption)</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
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.
- 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.

Table 3: SNMP Security Models and Levels, on page 6 identifies what the combinations of security models and levels mean.

Table 3: SNMP Security Models and Levels

<table>
<thead>
<tr>
<th>Model</th>
<th>Level</th>
<th>Authentication</th>
<th>Encryption</th>
<th>What Happens</th>
</tr>
</thead>
<tbody>
<tr>
<td>v1</td>
<td>noAuthNoPriv</td>
<td>Community string</td>
<td>No</td>
<td>Uses a community string match for authentication.</td>
</tr>
<tr>
<td>v2c</td>
<td>noAuthNoPriv</td>
<td>Community string</td>
<td>No</td>
<td>Uses a community string match for authentication.</td>
</tr>
<tr>
<td>v3</td>
<td>noAuthNoPriv</td>
<td>Username</td>
<td>No</td>
<td>Uses a username match for authentication.</td>
</tr>
<tr>
<td>v3</td>
<td>authNoPriv</td>
<td>HMAC-MD5 or HMAC-SHA</td>
<td>No</td>
<td>Provides authentication based on the HMAC-MD5 or HMAC-SHA.</td>
</tr>
</tbody>
</table>
### SNMPv3 Benefits

SNMPv3 provides secure access to devices by providing authentication, encryption and access control. These added security benefits secure SNMP against the following security threats:

- **Masquerade**—The threat that an SNMP user may assume the identity of another SNMP user to perform management operations for which that SNMP user does not have authorization.

#### Use of 3DES and AES Encryption Standards

Use of 3DES and AES encryption standards requires that the security package (k9sec) be installed. For information on installing software packages, see *Upgrading and Managing Cisco IOS XR Software*.

#### Table: Model, Level, Authentication, Encryption, What Happens

<table>
<thead>
<tr>
<th>Model</th>
<th>Level</th>
<th>Authentication</th>
<th>Encryption</th>
<th>What Happens</th>
</tr>
</thead>
<tbody>
<tr>
<td>v3</td>
<td>authPriv</td>
<td>HMAC-MD5 or HMAC-SHA</td>
<td>DES</td>
<td>Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES(^5) 56-bit encryption in addition to authentication based on the CBC(^5) DES (DES-56) standard.</td>
</tr>
<tr>
<td>v3</td>
<td>authPriv</td>
<td>HMAC-MD5 or HMAC-SHA</td>
<td>3DES</td>
<td>Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 168-bit 3DES(^6) level of encryption.</td>
</tr>
<tr>
<td>v3</td>
<td>authPriv</td>
<td>HMAC-MD5 or HMAC-SHA</td>
<td>AES</td>
<td>Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 128-bit AES(^7) level of encryption.</td>
</tr>
</tbody>
</table>

---

1. Hash-Based Message Authentication Code
2. Message Digest 5
3. Secure Hash Algorithm
4. Data Encryption Standard
5. Cipher Block Chaining
6. Triple Data Encryption Standard
7. Advanced Encryption Standard
• Message stream modification—The threat that messages may be maliciously reordered, delayed, or replayed (to an extent that is greater than can occur through the natural operation of a subnetwork service) to cause SNMP to perform unauthorized management operations.

• Disclosure—The threat that exchanges between SNMP engines could be eavesdropped. Protecting against this threat may be required as a matter of local policy.

In addition, SNMPv3 provides access control over protocol operations on SNMP managed objects.

**SNMPv3 Costs**

SNMPv3 authentication and encryption contribute to a slight increase in the response time when SNMP operations on MIB objects are performed. This cost is far outweighed by the security advantages provided by SNMPv3.

**Table 4: Order of Response Times from Least to Greatest, on page 8 shows the order of response time (from least to greatest) for the various security model and security level combinations.**

<table>
<thead>
<tr>
<th>Security Model</th>
<th>Security Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMPv2c</td>
<td>noAuthNoPriv</td>
</tr>
<tr>
<td>SNMPv3</td>
<td>noAuthNoPriv</td>
</tr>
<tr>
<td>SNMPv3</td>
<td>authNoPriv</td>
</tr>
<tr>
<td>SNMPv3</td>
<td>authPriv</td>
</tr>
</tbody>
</table>

**User-Based Security Model**

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.

USM uses two authentication protocols:

• HMAC-MD5-96 authentication protocol

• HMAC-SHA-96 authentication protocol
USM uses Cipher Block Chaining (CBC)-DES (DES-56) as the privacy protocol for message encryption.

View-Based Access Control Model

The View-Based Access Control Model (VACM) enables SNMP users to control access to SNMP managed objects by supplying read, write, or notify access to SNMP objects. It prevents access to objects restricted by views. These access policies can be set when user groups are configured with the `snmp-server group` command.

MIB Views

For security reasons, it is often valuable to be able to restrict the access rights of some groups to only a subset of the management information within the management domain. To provide this capability, access to a management object is controlled through MIB views, which contain the set of managed object types (and, optionally, the specific instances of object types) that can be viewed.

Access Policy

Access policy determines the access rights of a group. The three types of access rights are as follows:

- **read-view access**—The set of object instances authorized for the group when objects are read.
- **write-view access**—The set of object instances authorized for the group when objects are written.
- **notify-view access**—The set of object instances authorized for the group when objects are sent in a notification.

IP Precedence and DSCP Support for SNMP

SNMP IP Precedence and differentiated services code point (DSCP) support delivers QoS specifically for SNMP traffic. You can change the priority setting so that SNMP traffic generated in a router is assigned a specific QoS class. The IP Precedence or IP DSCP code point value is used to determine how packets are handled in weighted random early detection (WRED).

After the IP Precedence or DSCP is set for the SNMP traffic generated in a router, different QoS classes cannot be assigned to different types of SNMP traffic in that router.

The IP Precedence value is the first three bits in the type of service (ToS) byte of an IP header. The IP DSCP code point value is the first six bits of the differentiate services (DiffServ Field) byte. You can configure up to eight different IP Precedence markings or 64 different IP DSCP markings.

How to Implement SNMP on Cisco IOS XR Software

This section describes how to implement SNMP.

The `snmp-server` commands enable SNMP on Management Ethernet interfaces by default. For information on how to enable SNMP server support on other inband interfaces, see the Implementing Management Plane Protection on Cisco IOS XR Software module in Cisco IOS XR System Security Configuration Guide for the Cisco CRS Router.
Configuring SNMPv3

This task explains how to configure SNMPv3 for network management and monitoring.

Note
No specific command enables SNMPv3; the first `snmp-server` global configuration command (config), that you issue enables SNMPv3. Therefore, the sequence in which you issue the `snmp-server` commands for this task does not matter.

SUMMARY STEPS

1. configure
2. `snmp-server view view-name oid-tree {included | excluded}`
3. `snmp-server group name {v1 | v2c | v3 {auth | noauth | priv}} [read view] [write view] [notify view] [access-list-name]`
4. `snmp-server user username groupname {v1 | v2c | v3 {auth {md5 | sha} {clear | encrypted} auth-password [priv des56 {clear | encrypted} priv-password]} [access-list-name] [access-list-name]`
5. commit

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>`snmp-server view view-name oid-tree {included</td>
</tr>
<tr>
<td></td>
<td>Creates or modifies a view record.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>`snmp-server group name {v1</td>
</tr>
<tr>
<td></td>
<td>Configures a new SNMP group or a table that maps SNMP users to SNMP views.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>`snmp-server user username groupname {v1</td>
</tr>
<tr>
<td></td>
<td>Configures a new user to an SNMP group.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>commit</td>
</tr>
</tbody>
</table>
Configuring SNMP Trap Notifications

This task explains how to configure the router to send SNMP trap notifications.

Note: You can omit Step 3, on page 10 if you have already completed the steps documented under the Configuring SNMPv3, on page 10 task.

SUMMARY STEPS

1. configure
2. snmp-server group name {v1 | v2c | v3 {auth | noauth | priv}} [read view] [write view] [notify view] [access-list-name]
3. snmp-server user username groupname {v1 | v2c | v3 [auth {md5 | sha} {clear | encrypted} auth-password {priv des56 {clear | encrypted} priv-password}] [access-list-name]
4. snmp-server host address [traps] [version {1 | 2c | 3 [auth | noauth | priv]}] community-string [udp-port port] [notification-type]
5. snmp-server traps [notification-type]
6. commit
7. (Optional) show snmp host

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>configure</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>snmp-server group name {v1</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RP0/CPU0:router(config)# snmp-server group group_name v3 noauth read view_name1 write view_name2</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>snmp-server user username groupname {v1</td>
</tr>
<tr>
<td>Example:</td>
<td>RP/0/RP0/CPU0:router(config)# snmp-server user noauthuser group_name v3</td>
</tr>
</tbody>
</table>
Setting the Contact, Location, and Serial Number of the SNMP Agent

This task explains how to set the system contact string, system location string, and system serial number of the SNMP agent.

Note

The sequence in which you issue the `snmp-server` commands for this task does not matter.

**SUMMARY STEPS**

1. `configure`
2. (Optional) `snmp-server contact system-contact-string`
3. (Optional) `snmp-server location system-location`
4. (Optional) `snmp-server chassis-id serial-number`
5. `commit`
### Defining the Maximum SNMP Agent Packet Size

This task shows how to configure the largest SNMP packet size permitted when the SNMP server is receiving a request or generating a reply.

**Note**

The sequence in which you issue the `snmp-server` commands for this task does not matter.

**SUMMARY STEPS**

1. configure
2. (Optional) `snmp-server packetsize byte-count`
3. commit

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>configure</td>
</tr>
<tr>
<td>Step 2</td>
<td><code>snmp-server contact system-contact-string</code></td>
</tr>
<tr>
<td>Example:</td>
<td>```</td>
</tr>
<tr>
<td></td>
<td>RP/0/RP0/CPU0:router(config)# snmp-server contact</td>
</tr>
<tr>
<td></td>
<td>Dial System Operator at beeper # 27345</td>
</tr>
<tr>
<td>Step 3</td>
<td><code>snmp-server location system-location</code></td>
</tr>
<tr>
<td>Example:</td>
<td>```</td>
</tr>
<tr>
<td></td>
<td>RP/0/RP0/CPU0:router(config)# snmp-server location</td>
</tr>
<tr>
<td></td>
<td>Building 3/Room 214</td>
</tr>
<tr>
<td>Step 4</td>
<td><code>snmp-server chassis-id serial-number</code></td>
</tr>
<tr>
<td>Example:</td>
<td>```</td>
</tr>
<tr>
<td></td>
<td>RP/0/RP0/CPU0:router(config)# snmp-server chassis-id</td>
</tr>
<tr>
<td></td>
<td>1234456</td>
</tr>
<tr>
<td>Step 5</td>
<td>commit</td>
</tr>
</tbody>
</table>
### Implementing SNMP

#### Changing Notification Operation Values

After SNMP notifications have been enabled, you can specify a value other than the default for the source interface, message queue length, or retransmission interval.

This task explains how to specify a source interface for trap notifications, the message queue length for each host, and the retransmission interval.

**Note**

The sequence in which you issue the `snmp-server` commands for this task does not matter.

**SUMMARY STEPS**

1. `configure`
2. (Optional) `snmp-server trap-source type interface-path-id`
3. (Optional) `snmp-server queue-length length`
4. (Optional) `snmp-server trap-timeout seconds`
5. `commit`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> <code>snmp-server packetsize byte-count</code></td>
<td>(Optional) Sets the maximum packet size.</td>
</tr>
<tr>
<td><strong>Example:</strong> RP/0/RP0/CPU0:router(config)# snmp-server packetsize 1024</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>commit</code></td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td><strong>Step 3</strong> snmp-server queue-length length</td>
<td>(Optional) Establishes the message queue length for each notification.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# snmp-server queue-length 20</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> snmp-server trap-timeout seconds</td>
<td>(Optional) Defines how often to resend notifications on the retransmission queue.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>RP/0/RP0/CPU0:router(config)# snmp-server trap-timeout 20</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> commit</td>
<td></td>
</tr>
</tbody>
</table>

## Setting IP Precedence and DSCP Values

This task describes how to configure IP Precedence or IP DSCP for SNMP traffic.

**Before You Begin**

SNMP must be configured.

### SUMMARY STEPS

1. configure
2. Use one of the following commands:
   - `snmp-server ipv4 precedence value`
   - `snmp-server ipv4 dscp value`
3. commit

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> configure</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> Use one of the following commands:</td>
<td>Configures an IP precedence or IP DSCP value for SNMP traffic.</td>
</tr>
<tr>
<td>• <code>snmp-server ipv4 precedence value</code></td>
<td></td>
</tr>
<tr>
<td>• <code>snmp-server ipv4 dscp value</code></td>
<td></td>
</tr>
</tbody>
</table>
### Displaying SNMP Context Mapping

The SNMP agent serves queries based on SNMP contexts created by the client features. There is a context mapping table. Each entry in the context mapping table includes a context name, the name of the feature that created the context, and the name of the specific instance of the feature.

**SUMMARY STEPS**

1. `show snmp context-mapping`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>show snmp context-mapping</code></td>
<td>Displays the SNMP context mapping table.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>RP/0/RP0/CPU0:router# show snmp context-mapping</code></td>
<td></td>
</tr>
</tbody>
</table>

### Monitoring Packet Loss

It is possible to monitor packet loss by configuring the generation of SNMP traps when packet loss exceeds a specified threshold. The configuration described in this task enables the creation of entries in the MIB tables of the EVENT-MIB. This can then be monitored for packet loss using SNMP GET operations.

**Before You Begin**

**Note**

Entries created in the EVENT-MIB MIB tables using the configuration described in this task cannot be altered using an SNMP SET.

Entries to the EVENT-MIB MIB tables created using an SNMP SET cannot be altered using the configuration described in this task.
### SUMMARY STEPS

1. `snmp-server mibs eventmib packet-loss type interface-path-id falling lower-threshold interval sampling-interval rising upper-threshold`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>snmp-server mibs eventmib packet-loss type interface-path-id falling lower-threshold interval sampling-interval rising upper-threshold</code></td>
<td>Generates SNMP EVENT-MIB traps for the interface when the packet loss exceeds the specified thresholds. Up to 100 interfaces can be monitored.</td>
</tr>
</tbody>
</table>
| **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server mibs eventmib packet-loss falling 1 interval 5 rising 2` | *(lower-threshold)* — Specifies the lower threshold. When packet loss between two intervals falls below this threshold and an mteTriggerRising trap was generated previously, a SNMP mteTriggerFalling trap is generated. This trap is not generated until the packet loss exceeds the upper threshold and then falls back below the lower threshold.  
**interval sampling-interval** — Specifies how often packet loss statistics are polled. This is a value between 5 and 1440 minutes, in multiples of 5.  
**rising upper-threshold** — Specifies the upper threshold. When packet loss between two intervals increases above this threshold, an SNMP mteTriggereRising trap is generated. This trap is not generated until the packet loss drops below the lower threshold and then rises above the upper threshold. |

### Configuring MIB Data to be Persistent

Many SNMP MIB definitions define arbitrary 32-bit indices for their object tables. MIB implementations often do a mapping from the MIB indices to some internal data structure that is keyed by some other set of data. In these MIB tables the data contained in the table are often other identifiers of the element being modelled. For example, in the ENTITY-MIB, entries in the entPhysicalTable are indexed by the 31-bit value, entPhysicalIndex, but the entities could also be identified by the entPhysicalName or a combination of the other objects in the table.

Because of the size of some MIB tables, significant processing is required to discover all the mappings from the 32-bit MIB indices to the other data which the network management station identifies the entry. For this reason, it may be necessary for some MIB indices to be persistent across process restarts, switchovers, or device reloads. The ENTITY-MIB entPhysicalTable and CISCO-CLASS-BASED-QOS-MIB are two such MIBs that often require index values to be persistent.

Also, because of query response times and CPU utilization during CISCO-CLASS-BASED-QOS-MIB statistics queries, it is desirable to cache service policy statistics.
### SUMMARY STEPS

1. (Optional) `snmp-server entityindex persist`
2. (Optional) `snmp-server mibs cbqosmib persist`
3. (Optional) `snmp-server cbqosmib cache refresh time time`
4. (Optional) `snmp-server cbqosmib cache service-policy count count`
5. `snmp-server ifindex persist`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>snmp-server entityindex persist</code></td>
<td>(Optional) Enables the persistent storage of ENTITY-MIB data.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>RP/0/RP0/CPU0:router(config)# snmp-server entityindex persist</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>snmp-server mibs cbqosmib persist</code></td>
<td>(Optional) Enables persistent storage of the CISCO-CLASS-BASED-QOS-MIB data.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib persist</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><code>snmp-server cbqosmib cache refresh time time</code></td>
<td>(Optional) Enables QoS MIB caching with a specified cache refresh time.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache refresh time 45</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><code>snmp-server cbqosmib cache service-policy count count</code></td>
<td>(Optional) Enables QoS MIB caching with a limited number of service policies to cache.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache service-policy count 50</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td><code>snmp-server ifindex persist</code></td>
<td>Enables ifIndex persistence globally on all Simple Network Management Protocol (SNMP) interfaces.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>RP/0/RP0/CPU0:router(config)# snmp-server ifindex persist</code></td>
<td></td>
</tr>
</tbody>
</table>
Configuring LinkUp and LinkDown Traps for a Subset of Interfaces

By specifying a regular expression to represent the interfaces for which you are interested in setting traps, you can enable or disable linkUp and linkDown traps for a large number of interfaces simultaneously.

**Before You Begin**

SNMP must be configured.

**SUMMARY STEPS**

1. `configure`
2. `snmp-server interface subset subset-number regular-expression expression`
3. `notification linkupdown disable`
4. `commit`
5. (Optional) `show snmp interface notification subset subset-number`
6. (Optional) `show snmp interface notification regular-expression expression`
7. (Optional) `show snmp interface notification type interface-path-id`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 <code>configure</code></td>
<td>Enters snmp-server interface mode for the interfaces identified by the regular expression.</td>
</tr>
<tr>
<td>Step 2 <code>snmp-server interface subset subset-number regular-expression expression</code></td>
<td>The subset-number argument identifies the set of interfaces, and also assigns a priority to the subset in the event that an interface is included in more than one subset. Lower numbers have higher priority and their configuration takes precedence over interface subsets with higher numbers.</td>
</tr>
<tr>
<td>Example:</td>
<td>The <code>expression</code> argument must be entered surrounded by double quotes.</td>
</tr>
<tr>
<td>Step 3 <code>notification linkupdown disable</code></td>
<td>Disables linkUp and linkDown traps for all interfaces being configured. To enable previously disabled interfaces, use the <code>no</code> form of this command.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Step 4 <code>commit</code></td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>show snmp interface notification subset <strong>subset-number</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;RP/0/RP0/CPU0:router# show snmp interface notification subset 10</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>show snmp interface notification regular-expression <strong>expression</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;RP/0/RP0/CPU0:router# show snmp interface notification regular-expression &quot;^Gig[a-zA-Z]+[0-9/]+.&quot;</td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>show snmp interface notification <strong>type</strong> <strong>interface-path-id</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;RP/0/RP0/CPU0:router# show snmp interface notification tengige 0/4/0/3.10</td>
</tr>
</tbody>
</table>

### Configuration Examples for Implementing SNMP

### Configuring SNMPv3: Examples

#### Setting an Engine ID
This example shows how to set the identification of the local SNMP engine:

```
snmp-server engineID local 00:00:00:09:00:00:00:a1:61:6c:20:61
```

**Note**
After the engine ID has been configured, the SNMP agent restarts.

#### Verifying the Identification of the Local SNMP Engines
This example shows how to verify the identification of the local SNMP engine:

```
config
    show snmp engineid
```
Creating a View

There are two ways to create a view:

- You can include the object identifier (OID) of an ASN.1 subtree of a MIB family from a view by using the `included` keyword of the `snmp-server view` command.
- You can exclude the OID subtree of the ASN.1 subtree of a MIB family from a view by using the `excluded` keyword of the `snmp-server view` command.

This example shows how to create a view that includes the sysName (1.3.6.1.2.1.1.5) object:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 included
```

This example shows how to create a view that includes all the OIDs of a system group:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
```

This example shows how to create a view that includes all the OIDs under the system group except the sysName object (1.3.6.1.2.1.1.5), which has been excluded:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 excluded
```

Verifying Configured Views

This example shows how to display information about the configured views:

```
RP/0/RP0/CPU0:router# show snmp view
  v1default 1.3.6.1 - included nonVolatile active
  SNMP_VIEW1 1.3.6.1.2.1.1 - included nonVolatile active
  SNMP_VIEW1 1.3.6.1.2.1.1.5 - excluded nonVolatile active
```

Creating Groups

If you do not explicitly specify a notify, read, or write view, the Cisco IOS XR software uses the v1 default (1.3.6.1). This example shows how to create a group that utilizes the default view:

```
RP/0/RP0/CPU0:router(config)# snmp-server group group-name v3 auth
```

The following configuration example shows how to create a group that has read access to all the OIDs in the system except the sysUpTime object (1.3.6.1.2.1.1.3), which has been excluded from the view applied to the group, but write access only to the sysName object (1.3.6.1.2.1.1.5):

```
! snmp-server view view_name1 1.3.6.1.2.1.1 included
snmp-server view view_name1 1.3.6.1.2.1.1.3 excluded
snmp-server view view_name2 1.3.6.1.2.1.1.5 included
snmp-server group group_name1 v3 auth read view_name1 write view_name2
!```
Verifying Groups

This example shows how to verify the attributes of configured groups:

```plaintext
RP/0/RP0/CPU0:router# show snmp group

    groupname: group_name1
    readview : view_name1
    notifyview: v1default
    writeview: view_name2
    row status: nonVolatile
```

Creating and Verifying Users

Given the following SNMPv3 view and SNMPv3 group configuration:

```plaintext
!    snmp-server view view_name 1.3.6.1.2.1.1 included
    snmp-server group group_name v3 noauth read view_name write view_name
```

This example shows how to create a noAuthNoPriv user with read and write view access to a system group:

```plaintext
config
    snmp-server user noauthuser group_name v3
```

**Note**
The user must belong to a noauth group before a noAuthNoPriv user can be created.

This example shows how to verify the attributes that apply to the SNMP user:

```plaintext
RP/0/RP0/CPU0:router# show snmp user

    User name: noauthuser
    Engine ID: localSnmpID
    storage-type: nonvolatile active
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```plaintext
!    snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
    snmp-server group SNMP_GROUP1 v3 auth notify SNMP_VIEW1 read SNMP_VIEW1 write SNMP_VIEW1
```

This example shows how to create a user with authentication (including encryption), read, and write view access to a system group:

```plaintext
config
    snmp-server user userv3authpriv SNMP_GROUP1 v3 auth md5 password123 priv aes 128 password123
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```plaintext
!    snmp-server view view_name 1.3.6.1.2.1.1 included
    snmp group group_name v3 priv read view_name write view_name
```

This example shows how to create authNoPriv user with read and write view access to a system group:

```plaintext
RP/0/RP0/CPU0:router(config)# snmp-server user authuser group_name v3 auth md5 clear auth_passwd
```
Because the group is configured at a security level of Auth, the user must be configured as “auth” at a minimum to access this group (“priv” users could also access this group). The authNoPriv user configured in this group, authuser, must supply an authentication password to access the view. In the example, auth_passwd is set as the authentication password string. Note that clear keyword is specified before the auth_passwd password string. The clear keyword indicates that the password string being supplied is unencrypted.

This example shows how to verify the attributes that apply to SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user
User name: authuser
    Engine ID: localSnmpID
    storage-type: nonvolatile active
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
! snmp view view_name 1.3.6.1.2.1.1 included
snmp group group_name v3 priv read view_name write view_name
!
```

This example shows how to create an authPriv user with read and write view access to a system group:

```
config
snmp-server user privuser group_name v3 auth md5 clear auth_passwd priv des56 clear priv_passwd
```

Because the group has a security level of Priv, the user must be configured as a “priv” user to access this group. In this example, the user, privuser, must supply both an authentication password and privacy password to access the OIDs in the view.

This example shows how to verify the attributes that apply to the SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user
User name: privuser
    Engine ID: localSnmpID
    storage-type: nonvolatile active
```

### Configuring Trap Notifications: Example

The following example configures an SNMP agent to send out different types of traps. The configuration includes a v2c user, a noAuthNoPriv user, anAuthNoPriv user, and an AuthPriv user.

The default User Datagram Protocol (UDP) port is 161. If you do not specify a UDP port with the **udp-port** keyword and **port** argument, then the configured SNMP trap notifications are sent to port 161.
Implementing SNMP

Setting an IP Precedence Value for SNMP Traffic: Example

The following example shows how to set the SNMP IP Precedence value to 7:

```
configure
snmp-server ipv4 precedence 7
exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y
```

Setting an IP DSCP Value for SNMP Traffic: Example

The following example shows how to set the IP DSCP value of SNMP traffic to 45:

```
configure
```
snmp-server ipv4 dscp 45
exit

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y

## Additional References

The following sections provide references related to Implementing SNMP on Cisco IOS XR software.

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS XR SNMP commands</td>
<td>SNMP Server Commands on the Cisco IOS XR Software module of Cisco IOS XR System Management Command Reference for the Cisco CRS Router</td>
</tr>
<tr>
<td>MIB information</td>
<td>Cisco CRS and Cisco XR 12000 Series Router MIB Overview</td>
</tr>
<tr>
<td>Cisco IOS XR commands</td>
<td>Cisco IOS XR Commands Master List for the Cisco CRS Router</td>
</tr>
<tr>
<td>Getting started with Cisco IOS XR software</td>
<td>Cisco IOS XR Getting Started Guide for the Cisco CRS Router</td>
</tr>
<tr>
<td>Information about user groups and task IDs</td>
<td>Configuring AAA Services on the Cisco IOS XR Software module of Cisco IOS XR System Security Configuration Guide for the Cisco CRS Router</td>
</tr>
<tr>
<td>Cisco IOS XR Quality of Service</td>
<td>Cisco IOS XR Modular Quality of Service Configuration Guide for the Cisco CRS Router</td>
</tr>
</tbody>
</table>

### Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.</td>
<td>—</td>
</tr>
</tbody>
</table>
MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>—</td>
<td>To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: <a href="http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml">http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml</a></td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 3411</td>
<td><em>An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks</em></td>
</tr>
<tr>
<td>RFC 3412</td>
<td><em>Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)</em></td>
</tr>
<tr>
<td>RFC 3413</td>
<td><em>Simple Network Management Protocol (SNMP) Applications</em></td>
</tr>
<tr>
<td>RFC 3414</td>
<td><em>User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)</em></td>
</tr>
<tr>
<td>RFC 3415</td>
<td><em>View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)</em></td>
</tr>
<tr>
<td>RFC 3417</td>
<td><em>Transport Mappings for the Simple Network Management Protocol (SNMP)</em></td>
</tr>
<tr>
<td>RFC 3418</td>
<td><em>Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)</em></td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Implementing SNMP

Additional References