



Configuring SNMP

This chapter describes how to configure the Simple Network Management Protocol (SNMP) on your switch.



Note

For complete syntax and usage information for the commands used in this chapter, refer to the *Catalyst 3550 Multilayer Switch Command Reference* for this release and to the *Cisco IOS Configuration Fundamentals Command Reference for Release 12.1*.

This chapter consists of these sections:

- [Understanding SNMP, page 16-1](#)
- [Configuring SNMP, page 16-4](#)
- [Displaying SNMP Status, page 16-10](#)

Understanding SNMP

SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. The SNMP manager can be part of a network management system (NMS) such as CiscoWorks. The agent and management information base (MIB) reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent.

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

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), closing of a TCP connection, loss of connection to a neighbor, or other significant events.

This section includes this conceptual information:

- [SNMP Versions, page 16-2](#)
- [SNMP Manager Functions, page 16-2](#)
- [SNMP Agent Functions, page 16-3](#)
- [SNMP Community Strings, page 16-3](#)
- [Using SNMP to Access MIB Variables, page 16-3](#)

SNMP Versions

This software release supports these SNMP versions:

- SNMPv1—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- SNMPv2C, which has these features:
 - SNMPv2—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
 - SNMPv2C—The Community-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.

SNMPv2C replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the Community-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling 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.

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 software to support communications with one management station using the SNMPv1 protocol and another using the SNMPv2 protocol.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in [Table 16-1](#).

Table 16-1 *SNMP Operations*

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table. ¹
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent initiates this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent initiates this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of traps conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch.

A community string can have one of these attributes:

- Read-only—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access
- Read-write—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings
- Read-write-all—Gives read and write access to authorized management stations to all objects in the MIB, including the community strings



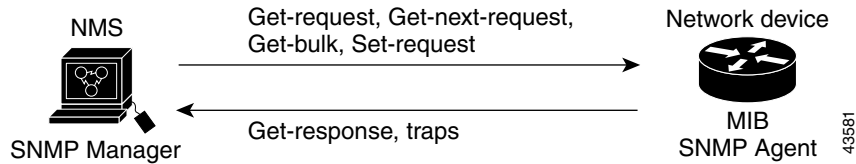
Note

When a cluster is created, the command switch manages the exchange of messages between member switches and the SNMP application. The Cluster Management software appends the member switch number (*@esN*, where *N* is the switch number) to the first configured RW and RO community strings on the command switch and propagates them to the member switches. For more information, see [Chapter 5, “Clustering Switches.”](#)

Using SNMP to Access MIB Variables

An example of an NMS is the CiscoWorks network management software. CiscoWorks 2000 software uses the switch MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in [Figure 16-1](#), the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps are messages alerting the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), and so forth. In addition, the SNMP agent responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 16-1 SNMP Network

For information on supported MIBs and how to access them, see [Appendix A, “Supported MIBs.”](#)

Configuring SNMP

This section describes how to configure SNMP on your switch. It contains this configuration information:

- [Default SNMP Configuration, page 16-4](#)
- [Configuring Community Strings, page 16-5](#)
- [Configuring Trap Managers and Enabling Traps, page 16-6](#)
- [Setting the Agent Contact and Location Information, page 16-8](#)
- [Limiting TFTP Servers Used Through SNMP, page 16-9](#)
- [SNMP Examples, page 16-9](#)

Default SNMP Configuration

[Table 16-2](#) shows the default SNMP configuration.

Table 16-2 Default SNMP Configuration

Feature	Default Setting
SNMP agent	Enabled
SNMP community strings	Read-Only: Public Read-Write: Private Read-Write-all: Secret
SNMP trap receiver	None configured
SNMP traps	None enabled

Disabling the SNMP Agent

Beginning in privileged EXEC mode, follow these steps to disable the SNMP agent:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	no snmp-server	Disable the SNMP agent operation.
Step 3	end	Return to privileged EXEC mode.
Step 4	show running-config	Verify your entries.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

No specific IOS command exists to enable SNMP. The first **snmp-server** command that you enter enables SNMPv1 and SNMPv2.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Beginning in privileged EXEC mode, follow these steps to configure a community string on the switch:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	snmp-server community <i>string</i> [ro rw] [<i>access-list-number</i>]	Configure the community string. For <i>string</i> , specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. (Optional) Specify either read-only (ro) if you want authorized management stations to retrieve MIB objects, or specify read-write (rw) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. (Optional) For <i>access-list-number</i> , enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.

	Command	Purpose
Step 3	<code>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</code>	<p>(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary.</p> <p>For <i>access-list-number</i>, enter the access list number specified in Step 2.</p> <p>The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched.</p> <p>For <i>source</i>, enter the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent.</p> <p>(Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore.</p> <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

**Note**

To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server community *string*** global configuration command.

This example assigns the string *comaccess* to SNMP allowing read-only access and specifies that IP access list 4 can use the community string to gain access to the switch SNMP agent:

```
Switch(config)# snmp-server community comaccess ro 4
```

Configuring Trap Managers and Enabling Traps

A trap manager is a management station that receives and processes traps. Traps are system alerts that the switch generates when certain events occur. By default, no trap manager is defined, and no traps are issued.

Switches running this IOS release can have an unlimited number of trap managers. Community strings can be any length.

Table 16-3 describes the supported switch traps (notification types). You can enable any or all of these traps and configure a trap manager to receive them.

Table 16-3 Switch Notification Types

Notification Type	Description
cluster	Generates a trap when the cluster configuration changes
config	Generates a trap for SNMP configuration changes
entity	Generates a trap for SNMP entity changes
HSRP	Generates a trap for Hot Standby Router Protocol (HSRP) changes
RTR	Generates a trap for the SNMP Response Time Reporter (RTR)
SNMP authentication	Generates a trap for SNMP authentication changes
TTY	Sends Cisco enterprise-specific notifications when a Transmission Control Protocol (TCP) connection closes
UDP-port	Sends notification of the User Datagram Protocol (UDP) port number of the host
VTP	Generates a trap for VLAN Trunk Protocol (VTP) changes

Some notification types cannot be controlled with the **snmp-server enable** command, for example, **tty** and **udp-port**. These notification types are always enabled. You can use the **snmp-server host** command to a specific host to receive the notification types listed in Table 16-3.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send traps to a host:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	snmp-server host <i>host-addr</i> [traps informs] [version { 1 2c }] <i>community-string notification-type</i>	<p>Specify the recipient of the trap message.</p> <p>For <i>host-addr</i>, specify the name or address of the host (the targeted recipient).</p> <p>(Optional) Specify traps (the default) to send SNMP traps to the host. Specify informs to send SNMP informs to the host.</p> <p>(Optional) Specify the SNMP version to support. Version 1, the default, is not available with informs.</p> <p>Note Though visible in the command-line help string, the version 3 keyword (SNMPv3) is not supported.</p> <p>For <i>community-string</i>, specify the string to send with the notification operation. Though you can set this string using the snmp-server host command, we recommend that you define this string by using the snmp-server community command before using the snmp-server host command.</p> <p>For <i>notification-type</i>, use the keywords listed in Table 16-3 on page 16-7.</p>

	Command	Purpose
Step 3	<code>snmp-server enable traps notification-types</code>	Enable the switch to send specific traps. For a list of traps, see Table 16-3 on page 16-7 . To enable multiple types of traps, you must issue a separate <code>snmp-server enable traps</code> command for each trap type.
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

To remove the specified host from receiving traps, use the `no snmp-server host host` global configuration command. To disable a specific trap type, use the `no snmp-server enable traps notification-types` global configuration command.

Setting the Agent Contact and Location Information

Beginning in privileged EXEC mode, follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file:

	Command	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>snmp-server contact text</code>	Set the system contact string. For example: <code>snmp-server contact Dial System Operator at beeper 21555.</code>
Step 3	<code>snmp-server location text</code>	Set the system location string. For example: <code>snmp-server location Building 3/Room 222</code>
Step 4	<code>end</code>	Return to privileged EXEC mode.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Beginning in privileged EXEC mode, follow these steps to limit the TFTP servers used for saving and loading configuration files by means of SNMP to the servers specified in an access list:

	Command	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	snmp-server tftp-server-list <i>access-list-number</i>	Limit TFTP servers used for configuration file copies through SNMP to the servers in the access list. For <i>access-list-number</i> , enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>]	Create a standard access list, repeating the command as many times as necessary. For <i>access-list-number</i> , enter the access list number specified in Step 2. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i> , enter the IP address of the TFTP servers that can access the switch. (Optional) For <i>source-wildcard</i> , enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	end	Return to privileged EXEC mode.
Step 5	show running-config	Verify your entries.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

SNMP Examples

This example enables SNMPv1 and SNMPv2C. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the switch to send any traps.

```
Switch(config)# snmp-server community public
```

This example permits any SNMP manager to access all objects with read-only permission using the community string *public*. The switch also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps vtp
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```

This example allows read-only access for all objects to members of access list 4 that specify the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

This example sends Entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the switch to send Entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server host** commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example enables the switch to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

Displaying SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. For information about the fields in this display, refer to the *Cisco IOS Configuration Fundamentals Command Reference for Release 12.1*.