

# Monitoring the Router and Network

---

This chapter describes the tasks that you can perform to monitor the router and network.

For a complete description of the router monitoring commands mentioned in this chapter, refer to the “Router and Network Monitoring Commands” chapter of the *Configuration Fundamentals Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

## Monitoring the Router and Network Task List

This chapter describes the tasks you can perform to manage the router and its performance on the network. Perform any of the tasks in the following sections:

- Configure SNMP Support
- Configure RMON Support
- Configure the Cisco Discovery Protocol
- Configure Response Time Reporter

## Configure SNMP Support

The Simple Network Management Protocol (SNMP) system consists of the following three parts:

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

SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents.

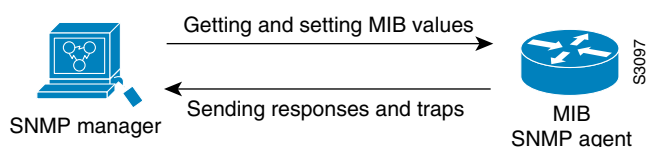
The SNMP manager can be part of a Network Management System (NMS) such as CiscoWorks. The agent and MIB reside on the router. To configure SNMP on the router, you define the relationship between the manager and the agent.

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 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 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 indicate improper user authentication, restarts, link status (up or down), closing of a TCP connection, loss of connection to a neighbor router, or other significant events.

Figure 357 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 traps to the manager to notify the manager of network conditions.

**Figure 357 Communication between an SNMP Agent and Manager**



## SNMP Notifications

The SNMP Inform Requests feature allows routers to send inform requests to SNMP managers.

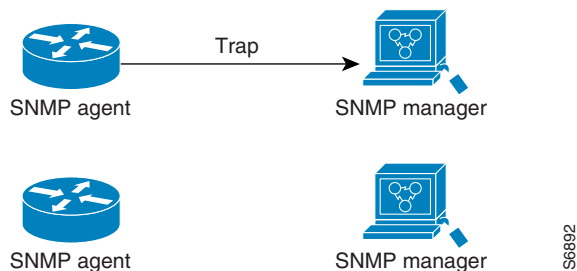
Routers can send notifications to SNMP managers when particular events occur. For example, an agent router might send a message to a manager when the agent router experiences an error condition.

SNMP notifications can be sent as traps or inform requests. Traps are unreliable because the receiver does not send any acknowledgment when it receives a trap. The sender cannot determine if the trap was received. However, an SNMP manager that receives an inform request acknowledges the message with an SNMP response 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.

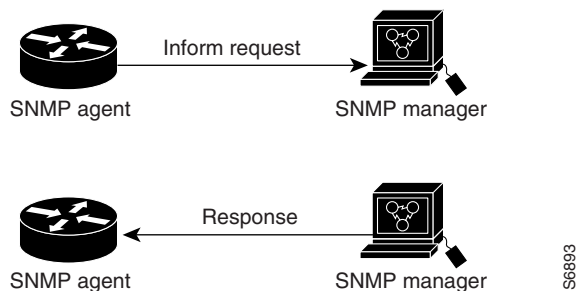
Because they are more reliable, informs consume more resources in the router and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once, while an inform may 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. If it is important that the SNMP manager receives every notification, use inform requests. On the other hand, if you are concerned about traffic on your network or memory in the router and you do not need to receive every notification, use traps.

Figure 358 through Figure 361 illustrate the differences between traps and inform requests.

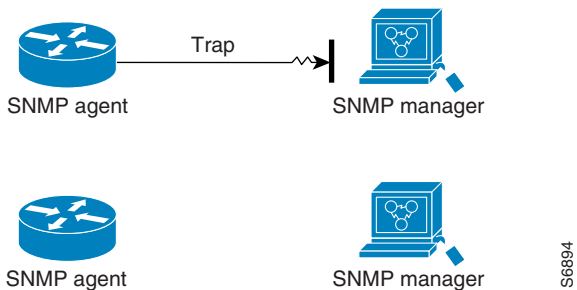
In Figure 358, the agent router successfully sends a trap to the SNMP manager. Although the manager receives the trap, it does not send any acknowledgment to the agent. The agent has no way of knowing that the trap reached its destination.

**Figure 358 Trap Sent to SNMP Manager Successfully**

In Figure 359, the agent router successfully sends an inform request to the manager. When the manager receives the inform request, it sends a response back to the agent. Thus, the agent knows that the inform request successfully reached its destination. Notice that, in this example, twice as much traffic is generated as in Figure 358; however, the agent is sure that the manager received the notification.

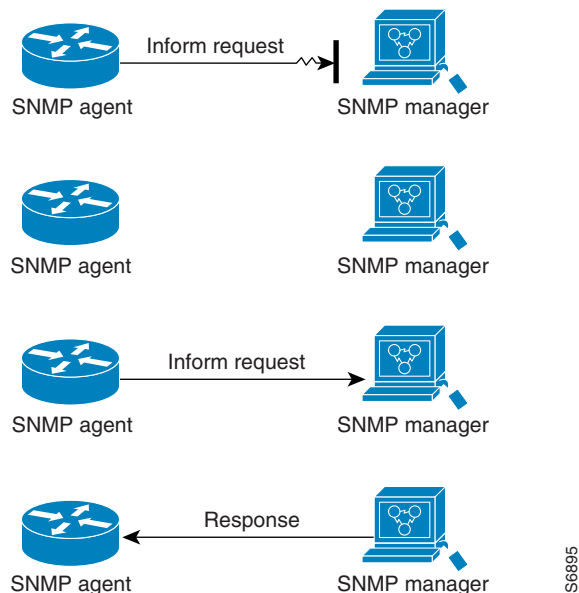
**Figure 359 Inform Request Sent to SNMP Manager Successfully**

In Figure 360, the agent sends a trap to the manager, but the trap does not reach the manager. Since 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 360 Trap Unsuccessfully Sent to SNMP Manager**

In Figure 361, the agent sends an inform request to the manager, but the inform request does not reach the manager. Since the manager did not receive the inform request, it does not send a response. After a period of time, the agent will resend the inform request. The second time, the manager receives the inform request and replies with a response. In this example, there is more traffic than in Figure 360; however, the notification reaches the SNMP manager.

**Figure 361 Inform Request Unsuccessfully Sent to SNMP Manager**



## Versions of SNMP

Cisco IOS Release 12.0 software supports the following versions of SNMP:

- **SNMPv1**—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- **SNMPv2C**, which consists of the following:
  - **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.

Cisco IOS Release 11.3 removed support for the following version of SNMP:

- **SNMPv2p (SNMPv2Classic)**—IETF Proposed Internet Standard of Version 2 of the Simple Network Management Protocol, defined in RFCs 1441 through 1451.

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 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.

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

## Supported MIBs

For a complete listing of supported MIBs by platform and release, see the “Cisco MIBs” section of Cisco Connection Online at <http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

For a listing of supported trap and inform types, see the **snmp-server enable traps** command description in the *Configuration Fundamentals Command Reference*.

## SNMP Configuration Task List

There is no specific command that you use to enable SNMP. The first **snmp-server** command that you enter enables both versions of SNMP.

To configure SNMP support, perform any of the tasks in the following sections. The second task is required; all other tasks are optional.

- Create or Modify an SNMP View Record
- Create or Modify Access Control for an SNMP Community
- Enable the SNMP Agent Shutdown Mechanism
- Establish the Contact, Location, and Serial Number of the SNMP Agent
- Define the Maximum SNMP Agent Packet Size
- Limit TFTP Servers Used Via SNMP
- Monitor SNMP Status
- Disable the SNMP Agent
- Configure SNMP Traps
- Configure SNMP Informs
- Configure the Router as an SNMP Manager

## Create or Modify an SNMP View Record

You can assign views to community strings to limit which MIB objects an SNMP manager can access. You can use a predefined view, or create your own view. If you are using a predefined view or no view at all, skip this task.

To create or modify an SNMP view record, use the following command in global configuration mode:

Command	Purpose
<b>snmp-server view</b> <i>view-name oid-tree</i> { <b>included</b>   <b>excluded</b> }	Create or modify a view record.

To remove a view record, use the **no snmp-server view** command.

You can enter this command multiple times for the same view record. Later lines take precedence when an object identifier is included in two or more lines.

## Create or Modify Access Control for an SNMP Community

Use an 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 router. Optionally, you can specify one or more of the following 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.

To configure a community string, use the following command in global configuration mode:

Command	Purpose
<code>snmp-server community string [view view-name] [ro   rw] [number]</code>	Define the community access string.

You can configure one or more community strings. To remove a specific community string, use the **no snmp-server community** command.

For an example of configuring a community string, see the section “SNMP Examples” at the end of this chapter.

## Enable the SNMP Agent Shutdown Mechanism

Using SNMP packets, a network management tool can send messages to users on virtual terminals and the console. This facility operates in a similar fashion to the EXEC **send** command; however, the SNMP request that causes the message to be issued to the users also specifies the action to be taken after the message is delivered. One possible action is a shutdown request. After a system is shut down, typically it is reloaded. Because the ability to cause a reload from the network is a powerful feature, it is protected by the **snmp-server system-shutdown** global configuration command. If you do not issue this command, the shutdown mechanism is not enabled. To enable the SNMP agent shutdown mechanism, use the following command in global configuration mode:

Command	Purpose
<code>snmp-server system-shutdown</code>	Use the SNMP message reload feature and request a system shutdown message.

To understand how to use this feature with SNMP requests, read the document `OLD-CISCO-SYSTEM-MIB.my`, available on Cisco Connection Online.

## Establish the Contact, Location, and Serial Number of the SNMP Agent

You can set the system contact, location, and serial number of the SNMP agent so that these descriptions can be accessed through the configuration file. To do so, use one or more of the following commands in global configuration mode:

Command	Purpose
<b>snmp-server contact</b> <i>text</i>	Set the system contact string.
<b>snmp-server location</b> <i>text</i>	Set the system location string.
<b>snmp-server chassis-id</b> <i>number</i>	Set the system serial number.

## Define the Maximum SNMP Agent Packet Size

You can set the maximum packet size permitted when the SNMP agent is receiving a request or generating a reply. To do so, use the following command in global configuration mode:

Command	Purpose
<b>snmp-server packetsize</b> <i>byte-count</i>	Establish the maximum packet size.

## Limit TFTP Servers Used Via SNMP

You can limit the TFTP servers used for saving and loading configuration files via SNMP to the servers specified in an access list. To do so, use the following command in global configuration mode:

Command	Purpose
<b>snmp-server tftp-server-list</b> <i>number</i>	Limit TFTP servers used for configuration file copies via SNMP to the servers in an access list.

## Monitor SNMP Status

To monitor SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the following command in EXEC mode:

Command	Purpose
<b>show snmp</b>	Monitor SNMP status.

## Disable the SNMP Agent

To disable both versions of SNMP (SNMPv1 and SNMPv2C), use the following command in global configuration mode:

Command	Purpose
<b>no snmp-server</b>	Disable SNMP agent operation.

## Configure SNMP Traps

To configure the router to send SNMP traps, use the following commands. The second task is optional.

- Configure the Router to Send Traps
- Change Trap Operation Values

### Configure the Router to Send Traps

To configure the router to send traps to a host, use the following commands in global configuration mode:

Step	Command	Purpose
1	<b>snmp-server host</b> <i>host</i> [ <b>version</b> {1   2c}] <i>community-string</i> [ <b>udp-port</b> <i>port</i> ] [ <i>notification-type</i> ]	Specify the recipient of the trap message.
2	<b>snmp-server enable traps</b> [ <i>notification-type</i> ] [ <i>notification-option</i> ]	Specify the types of traps sent. This command also specifies which types of informs are enabled.

The **snmp-server host** command specifies which hosts will receive traps. The **snmp-server enable traps** command globally enables the trap production mechanism for the specified traps. In order for a host to receive a trap, an **snmp-server host** command must be configured for that host, and, generally, the trap must be enabled globally through the **snmp-server enable traps** command.

Note, however, that some traps and informs are not controlled by the **snmp-server enable traps** command. These traps are either enabled by default or controlled through other commands.

Other traps and informs can be controlled in conjunction with other commands. For example, SNMP linkUp and linkDown notifications are enabled by issuing the **snmp-server enable traps snmp** command in global configuration mode. However, you can disable these traps on specific interfaces using the **no snmp trap link-status** command in interface configuration mode. While linkUp and linkDown traps are enabled by default for individual interfaces, these traps will not be sent unless you issue the **snmp-server enable traps snmp [authentication]** command in global configuration mode. (The ability to disable link-status traps for individual interfaces allows you to disable these traps for interfaces expected to go up and down during normal usage, such as ISDN interfaces.)

---

**Note** The **snmp-server enable traps snmp [authentication]** global configuration command enables the following notification types: *authentication failure*, *linkUp*, *linkDown*, and *coldstart*. However, only the “authentication” keyword will appear in the configuration file. See RFC 1157 for a definition of these notification types.

---

### Change Trap Operation Values

Optionally, you can specify a value other than the default for the source interface, message (packet) queue length for each host, or retransmission interval.

To change trap operation values, use any of the following optional commands in global configuration mode:

Command	Purpose
<b>snmp-server trap-source</b> <i>interface</i>	Specify the source interface (and hence IP address) of the trap message. This command also sets the source IP address for informs.

Command	Purpose
<b>snmp-server queue-length</b> <i>length</i>	Establish the message queue length for each trap host.
<b>snmp-server trap-timeout</b> <i>seconds</i>	Define how often to resend trap messages on the retransmission queue.

## Configure SNMP Informs

To configure the router to send SNMP informs, use the following commands. The second task is optional.

- Configure the Router to Send Informs
- Change Inform Operation Values

### Configure the Router to Send Informs

To configure the router to send informs to a host, use the following commands in global configuration mode:

Command	Purpose
<b>snmp-server host</b> <i>host</i> <b>informs</b> [ <b>version 2c</b> ] <i>community-string</i> [ <b>udp-port</b> <i>port</i> ] [ <i>notification-type</i> ]	Specify the recipient of the inform message.
<b>snmp-server enable traps</b> [ <i>notification-type</i> ] [ <i>notification-option</i> ]	Specify the types of inform requests sent. This command also specifies which types of traps are enabled.

The **snmp-server host** command specifies which hosts will receive informs. The **snmp-server enable traps** command globally enables the production mechanism for the specified notifications (traps and informs).

Some informs are not controlled by the **snmp-server enable traps** command. These informs are either enabled by default or controlled through other commands.

In order for a host to receive an inform, an **snmp-server host informs** command must be configured for that host, and the inform must be enabled globally through the **snmp-server enable traps** command (except for informs that are enabled through a different command, or informs that are enabled by default).

### Change Inform Operation Values

Optionally, you can specify a value other than the default for number of retries, the retransmission interval, the maximum number of pending requests, or the source IP address.

To change inform operation values, use the following optional command in global configuration mode:

Step	Command	Purpose
1	<b>snmp-server informs</b> [ <b>retries</b> <i>retries</i> ] [ <b>timeout</b> <i>seconds</i> ] [ <b>pending</b> <i>pending</i> ]	Set options related to resending unacknowledged inform requests.
2	<b>snmp-server trap-source</b> <i>interface</i>	Specify the source interface (and hence IP address) of the inform request. This command also changes the source interface for traps.

## Configure the Router as an SNMP Manager

The SNMP Manager feature allows a router to serve as an SNMP manager. As an SNMP manager, the router can send SNMP requests to agents and receive SNMP responses and notifications from agents. When the SNMP manager process is enabled, the router can query other SNMP agents and process incoming SNMP traps.

### Security Considerations

Most network security policies assume that routers will be accepting SNMP requests, sending SNMP responses, and sending SNMP notifications.

With the SNMP manager functionality enabled, the router may also be sending SNMP requests, receiving SNMP responses, and receiving SNMP notifications. Your security policy implementation may need to be updated prior to enabling this feature.

SNMP requests are typically sent to UDP port 161. SNMP responses are typically sent from UDP port 161. SNMP notifications are typically sent to UDP port 162.

### SNMP Sessions

Sessions are created when the SNMP manager in the router sends SNMP requests, such as inform requests, to a host or receives SNMP notifications from a host. One session is created for each destination host. If there is no further communication between the router and host within the session timeout period, the session will be deleted.

The router tracks statistics, such as the average round-trip time required to reach the host, for each session. Using the statistics for a session, the SNMP manager in the router can set reasonable timeout periods for future requests, such as informs, for that host. If the session is deleted, all statistics are lost. If another session with the same host is later created, the request timeout value for replies will return to the default value.

Sessions consume memory. A reasonable session timeout value should be large enough that regularly used sessions are not prematurely deleted, yet small enough such that irregularly used, or one-shot sessions, are purged expeditiously.

### Configuration Tasks

To configure the router to act as an SNMP manager, use the tasks in the following sections:

- Enable the SNMP Manager
- Monitor the SNMP Manager

### Enable the SNMP Manager

To enable the SNMP manager process and optionally set the session timeout value, use the following commands in global configuration mode:

Step	Command	Purpose
1	<code>snmp-server manager</code>	Enable the SNMP Manager.
2	<code>snmp-server manager session-timeout seconds</code>	(Optional) Change the session timeout value.

## Monitor the SNMP Manager

To monitor the SNMP manager process, use any one of the following commands in EXEC mode:

Step	Command	Purpose
1	<code>show snmp</code>	Display global SNMP information.
2	<code>show snmp sessions [brief]</code>	Display information about current sessions.
3	<code>show snmp pending</code>	Display information about current pending requests.

## Configure RMON Support

The Remote Monitoring (RMON) option provides visibility of individual nodal activity and allows you to monitor all nodes and their interaction on a LAN segment. RMON, used in conjunction with the SNMP agent in the router, allows you to view both traffic that flows through the router and segment traffic not necessarily destined for the router. Combining RMON alarms and events with existing MIBs allows you to choose where proactive monitoring will occur.

Full RMON packet analysis as described in RFC 1757 is available only on an Ethernet interface of the Cisco 2500 series and Cisco AS5200 series routers. RMON requires that SNMP be configured. A generic RMON console application is recommended in order to take advantage of RMON's network management capabilities.

RMON can be very data and processor intensive. Users should measure usage effects to ensure that router performance is not degraded and to minimize excessive management traffic overhead. Native mode is less intensive than promiscuous mode.

All Cisco IOS software images ordered without the explicit RMON option include limited RMON support (RMON alarms and event groups only). Images ordered with the RMON option include support for all nine groups (statistics, history, alarms, hosts, hostTopN, matrix, filter, capture, and event). As a security precaution, support for the packet capture group allows capture of packet header information only; data payloads are not captured.

To enable RMON on an Ethernet interface, use the following command in interface configuration mode:

Command	Purpose
<code>rmon {native   promiscuous}</code>	Enable RMON.

In native mode, RMON monitors only the packets normally received by the interface. In promiscuous mode, RMON monitors all packets on the LAN segment.

The default size of the queue that holds packets for analysis by the RMON process is 64 packets. To change the size of the queue, use the following command in global configuration mode:

Command	Purpose
<code>rmon queuesize size</code>	Change the size of the RMON queue.

To set an RMON alarm or event, use one of the following commands in global configuration mode:

Command	Purpose
<code>rmon alarm number variable interval {delta   absolute} rising-threshold value [event-number] falling-threshold value [event-number] [owner string]</code>	Set an alarm on a MIB object.

Command	Purpose
<b>rmon event</b> <i>number</i> [ <b>log</b> ] [ <b>trap</b> <i>community</i> ] [ <b>description</b> <i>string</i> ] [ <b>owner</b> <i>string</i> ]	Add or remove an event in the RMON event table.

You can set an alarm on any MIB object in the access server. To disable an alarm, you must enable the **no** form of this command on each alarm you configure. You cannot disable all the alarms you configure at once. Refer to RFC 1757 to learn more about alarms and events and how they interact with each other.

To display the current RMON status, use one or more of the following commands in EXEC mode:

Command	Purpose
<b>show rmon</b>	Display general RMON statistics.
or	
<b>show rmon task</b>	
<b>show rmon alarms</b>	Display the RMON alarm table.
<b>show rmon capture</b>	Display the RMON buffer capture table. Available on Cisco 2500 series and Cisco AS5200 only.
<b>show rmon events</b>	Display the RMON event table.
<b>show rmon filter</b>	Display the RMON filter table. Available on Cisco 2500 series and Cisco AS5200 only.
<b>show rmon history</b>	Display the RMON history table. Available on Cisco 2500 series and Cisco AS5200 only.
<b>show rmon hosts</b>	Display the RMON hosts table. Available on Cisco 2500 series and Cisco AS5200 only.
<b>show rmon matrix</b>	Display the RMON matrix table. Available on Cisco 2500 series and Cisco AS5200 only.
<b>show rmon statistics</b>	Display the RMON statistics table. Available on Cisco 2500 series and Cisco AS5200 only.
<b>show rmon topn</b>	Display the RMON top-n hosts table. Available on Cisco 2500 series and Cisco AS5200 only.

For an example of configuring RMON alarms and events, see the section “RMON Alarm and Event Examples” at the end of this chapter.

## Configure the Cisco Discovery Protocol

The Cisco Discovery Protocol (CDP) is media- and protocol-independent, and runs on all Cisco-manufactured equipment including routers, bridges, access servers, and switches. With CDP, network management applications can learn the device type and the SNMP agent address of neighboring devices. This enables applications to send SNMP queries to neighboring devices.

CDP runs on all media that support Subnetwork Access Protocol (SNAP), including local-area network (LAN), Frame Relay, and Asynchronous Transfer Mode (ATM) media. CDP runs over the data link layer only. Therefore, two systems that support different network-layer protocols can learn about each other.

Each device configured for CDP sends periodic messages to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime, information, which indicates the length of time a receiving device should hold CDP information before discarding it.

There is a CDP MIB for the management of CDP on Cisco devices.

## CDP Configuration Task List

To configure CDP, perform the tasks in the following sections:

- Set the CDP Transmission Timer and Hold Time
- Enable CDP
- Enable CDP on an Interface
- Monitor and Maintain CDP

---

**Note** The **cdp enable**, **cdp timer**, and **cdp run** commands affect the operation of the IP on demand routing feature (that is, the **router odr** global configuration command). For more information on the **router odr** command, see the “On-Demand Routing Commands” chapter in the *Network Protocols Command Reference, Part 1*.

---

### Set the CDP Transmission Timer and Hold Time

To set the frequency of CDP transmissions and the hold time for CDP packets, use the following commands in global configuration mode:

Step	Command	Purpose
1	<b>cdp timer</b> <i>seconds</i>	Specify frequency of transmission of CDP updates.
2	<b>cdp holdtime</b> <i>seconds</i>	Specify the amount of time a receiving device should hold the information sent by your device before discarding it.

### Enable CDP

CDP is enabled by default. If you prefer not to use the CDP device discovery capability, you can disable it with the **no cdp run** command.

To reenable CDP after disabling it, use the following command in global configuration mode:

Command	Purpose
<b>cdp run</b>	Enable CDP.

### Enable CDP on an Interface

CDP is enabled by default on all supported interfaces to send and receive CDP information. However, some interfaces, such as ATM interfaces, do not support CDP. You can disable CDP on an interface which supports CDP with the **no cdp enable** command.

To reenabling CDP on an interface after disabling it, use the following command in interface configuration mode:

Command	Purpose
<b>cdp enable</b>	Enable CDP on an interface.

## Monitor and Maintain CDP

To monitor and maintain CDP on your device, use one or more of the following commands in privileged EXEC mode:

Command	Purpose
<b>clear cdp counters</b>	Reset the traffic counters to zero.
<b>clear cdp table</b>	Delete the CDP table of information about neighbors.
<b>show cdp</b>	Display global information such as frequency of transmissions and the holdtime for packets being transmitted.
<b>show cdp entry</b> <i>entry-name</i> [ <b>protocol</b>   <b>version</b> ]	Display information about a specific neighbor. Display can be limited to protocol or version information.
<b>show cdp interface</b> [ <i>type number</i> ]	Display information about interfaces on which CDP is enabled.
<b>show cdp neighbors</b> [ <i>type number</i> ] [ <b>detail</b> ]	Display information about neighbors. The display can be limited to neighbors on a specific interface, and expanded to provide more detailed information.
<b>show cdp traffic</b>	Display CDP counters, including the number of packets sent and received and checksum errors.
<b>show debugging</b>	Display information about the types of debugging that are enabled for your router. See the <i>Debug Command Reference</i> for more information about CDP <b>debug</b> commands.

## Configure Response Time Reporter

The response time reporter feature allows you to monitor network performance, network resources, and applications by measuring response times and availability. With this feature you can perform troubleshooting, problem notifications, and preproblem analysis using response time reporter statistics as a baseline.

The response time reporter feature is currently available only with the IBM feature set of the Cisco IOS software. A CiscoWorksBlue network management application will be available to support the response time reporter feature. Both the CiscoWorks Blue network management application and the router use the Cisco Round Trip Time Monitor (RTTMON) MIB.

You can use the response time reporter feature to troubleshoot problems by checking the time delays between devices (such as a router and an MVS host), and the time delays on the path from the source device to the destination device at the protocol level.

You can also use this feature to send any combination of SNMP traps and SNA Alerts/Resolutions when one of the following has occurred: a user-configured threshold is exceeded, a connection is lost and reestablished, or when a timeout occurs. Thresholds can also be used to trigger additional collection of time delay statistics.

You can use this feature to perform preproblem analysis by scheduling the response time reporter and collecting the results as history and accumulated statistics. You can then use the statistics to model and predict future network topologies.

## Response Time Reporter Configuration Task List

To configure the response time reporter feature, complete the tasks in the following sections. Configuring the probe and scheduling the probe are required tasks; the remaining tasks are optional.

- Configure the Probe
- Capture Statistics and Collect Error Information
- Collect History
- Set Reaction Conditions
- Schedule the Probe
- Reset the Probe
- Monitor the Response Time Reporter Feature

See the end of this chapter for “Response Time Reporter Examples.”

### Configure the Probe

Response time and availability information is collected by *probes* (devices specifically placed in a network to collect data about the network) that you configure on the router. To configure a new response time reporter probe, use the following commands starting in global configuration mode:

Step	Command	Purpose
1	<b>rtr probe</b>	Enter response time reporter configuration mode.
2	<b>type {echo   pathecho} protocol type type-target</b>	Specify the type of probe.

You must configure the probe’s type before you can configure any of the other characteristics.

---

**Note** When the probe type is **pathEcho**, statistics are recorded for each hop along the path that the probe takes to reach its destination.

---

To configure optional characteristics, use the following commands in response time reporter configuration mode:

Step	Command	Purpose
1	<b>frequency seconds</b>	Set the rate at which the probe starts a response time reporter operation.
2	<b>owner text</b>	Configure the SNMP owner of the probe.
3	<b>threshold milliseconds</b>	Set the rising threshold (hysteresis) that generates a reaction event and stores history information for the probe.
4	<b>timeout milliseconds</b>	Set the amount of time the probe waits for a response from its request packet.
5	<b>request-data-size bytes</b>	Set the protocol data size in the payload of the probe’s request packet.
6	<b>response-data-size bytes</b>	Set the protocol data size in the payload of the probe’s response packet.
7	<b>tag text</b>	Logically link probes together in a group.

Step	Command	Purpose
8	<code>verify-data</code>	Check each probe response for corruption.

## Capture Statistics and Collect Error Information

The main purpose of the probe is to capture statistics and collect error information. By default, the following information is captured and collected:

- Minimum and maximum response times
- Number of completions
- Sum of completion times
- Sum of the squares of completion times
- Accumulation of errors for noncompletions
- Total attempts (errors plus number of completions)
- Statistical distributions of response times

In most situations, you do not need to change the statistical distribution interval or size. Only change the size when distributions are needed (for example, when performing statistical modeling of your network).

To control how much and what type of statistics are stored on the router, use the following optional commands in response time reporter configuration mode:

Command	Purpose
<code>statistics-distribution-interval</code> <i>milliseconds</i>	Set the time interval for each statistical distribution kept.
<code>distributions-of-statistics-kept</code> <i>size</i>	Set number of statistical distributions kept per hop during the probe's lifetime.
<code>hops-of-statistics-kept</code> <i>size</i>	Set the number of hops for which statistics are maintained per path for the probe.
<code>paths-of-statistics-kept</code> <i>size</i>	Set the number of paths for which statistics are maintained per hour for the probe.
<code>hours-of-statistics-kept</code> <i>hours</i>	Set the number of hours for which statistics are maintained for the probe.

---

**Note** When using a distribution size of 1 (the default), you do not need to set the `statistics-distribution-interval` response time reporter configuration command because it has no effect on the statistics kept. For more information, refer to the command in the “Router and Network Monitoring Commands” chapter of the *Configuration Fundamentals Command Reference*.

---

## Collect History

A probe can collect history and capture statistics. By default, history is not collected. When a problem arises where history is useful (for example, a large number of timeouts are occurring), you can configure the probe to collect history.

---

**Note** Collecting history increases the RAM usage. Only collect history when you think there is a problem. For general network response time information, use statistics.

---

To control how much and what type of history is stored on the router, use the following commands in response time reporter configuration mode. The first command is required; the remainder are optional.

Step	Command	Purpose
1	<b>samples-of-history-kept</b> <i>samples</i>	Set the number of entries kept in the history table per bucket.
2	<b>buckets-of-history-kept</b> <i>size</i>	Set the number of history buckets that are kept per lives-of-history-kept.
3	<b>lives-of-history-kept</b> <i>lives</i>	Enable history collection and set the number of lives maintained in the history table for the probe.
4	<b>filter-for-history</b> { <b>none</b>   <b>all</b>   <b>overthreshold</b>   <b>failures</b> }	Define the type of information kept in the history table for the probe.

To disable history collection, use the default value (0 lives) for the **lives-of-history-kept** command rather than the **filter-for-history none** response time reporter configuration command. The **lives-of-history-kept** command disables history collection before the probe's operation is attempted, and the **filter-for-history** command with the **none** keyword checks for history inclusion after the probe's operation attempt is made.

## Set Reaction Conditions

You can configure the probe to send threshold notifications and use those notifications to trigger additional collection of time delay statistics. You can also configure the probe to send notifications when the probe loses connection, reestablishes connections, times out, and first succeeds after a timeout.

To configure the probe's reaction conditions, use the following optional commands in global configuration mode:

Step	Command	Purpose
1	<b>rtr reaction-configuration</b> <i>probe</i> [ <b>connection-loss-enable</b> ] [ <b>timeout-enable</b> ] [ <b>threshold-falling</b> <i>milliseconds</i> ] [ <b>threshold-type</b> <i>option</i> ] [ <b>action-type</b> <i>option</i> ]	Configure certain actions to occur based on events under the control of the response time reporter.
2	<b>rtr reaction-trigger</b> <i>probe target-probe</i>	Define the target probe to make the transition from a "pending" state to an "active" state when one of the trigger action-type options is defined for the probe.

## Schedule the Probe

After you have configured the probe, you must schedule the probe to begin capturing statistics and collecting error information. To do so, use the following command in global configuration mode:

Command	Purpose
<b>rtr schedule</b> <i>probe</i> [ <b>life</b> <i>seconds</i> ] [ <b>start-time</b> { <b>pending</b>   <b>now</b>   <i>hh:mm</i> [ <i>month day</i>   <i>day month</i> ]}] [ <b>ageout</b> <i>seconds</i> ]	Schedule the probe by configuring the time parameters.

---

**Note** After you schedule the probe with the **rtr schedule** command, you cannot change the probe's configuration with the **rtr** global configuration command. To change the configuration of a probe that has been scheduled, use the **no** form of the **rtr** command. The **no** form removes all the probe's configuration information including the probe's schedule, reaction configuration, and reaction triggers. You can now create a new configuration for the probe.

---

If the probe is in a pending state (the default), you can define the conditions under which the probe makes the transition from pending to active with the **rtr reaction-trigger** global configuration command. When the probe is in an active state it immediately begins collecting information.

## Reset the Probe

To perform a shutdown and restart of the response time reporter, use the following command in global configuration mode:

Command	Purpose
<b>rtr reset</b>	Stop all probes and clear the response time reporter configuration information.



**Caution** Use the **rtr reset** command only in extreme situations such as the incorrect configuration of a number of probes.

In addition to stopping all probes and clearing the response time reporter configuration information, the **rtr reset** command returns the response time reporter feature to the startup condition. This command does not reread the configuration stored in NVRAM. You must retype the response time reporter's configuration or use the **config memory** command (this has the side effect of reconfiguring the router to its startup configuration).

## Monitor the Response Time Reporter Feature

To display information about the status and configuration of the response time reporter feature, use the following commands in EXEC mode. You can display information in a tabular or full format. Tabular format displays information in a column reducing the number of screens required to display the information. Full format displays all information using identifiers next to each displayed value.

Command	Purpose
<b>show rtr application</b> [ <b>tabular</b>   <b>full</b> ]	Display global information about the response time reporter feature.
<b>show rtr collection-statistics</b> [ <i>probe</i> ] [ <b>tabular</b>   <b>full</b> ]	Display error totals collected for all probes or the specified probe.

Command	Purpose
<code>show rtr configuration [probe] [tabular   full]</code>	Display configuration values including all defaults for all probes or the specified probe.
<code>show rtr distribution-statistics [probe] [tabular   full]</code>	Display statistical distribution information (captured response times) for all probes or the specified probe.
<code>show rtr history [probe] [tabular   full]</code>	Display history collected for all probes or the specified probe.
<code>show rtr operational-state [probe] [tabular   full]</code>	Display the operational state of all probes or the specified probe.
<code>show rtr reaction-trigger [probe] [tabular   full]</code>	Display the reaction trigger information for all probes or the specified probe.
<code>show rtr totals-statistics [probe] [tabular   full]</code>	Display the total statistic values (accumulation of error counts and completions) for all probes or the specified probe.

## Monitor the Router and Network Configuration Examples

The following sections provide system management examples:

- SNMP Examples
- RMON Alarm and Event Examples
- Response Time Reporter Examples

### SNMP Examples

The following 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 router to send any traps.

```
snmp-server community public
```

The following example permits any SNMP to access all objects with read-only permission using the community string public. The router will also send ISDN 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.

```
snmp-server community public
snmp-server enable traps isdn
snmp-server host 192.180.1.27 version 2c public
snmp-server host 192.180.1.111 version 1 public
snmp-server host 192.180.1.33 public
```

The following 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.

```
snmp-server community comaccess ro 4
snmp-server enable traps snmp authentication
snmp-server host cisco.com version 2c public
```

The following example sends Entity MIB traps to the host cisco.com. The community string is restricted. The first line enables the router 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.

```
snmp-server enable traps entity
snmp-server host cisco.com restricted entity
```

The following example sends the SNMP and Cisco environmental monitor enterprise-specific traps to address 172.30.2.160:

```
snmp-server enable traps
snmp-server host 172.30.2.160 public snmp envmon
```

The following example enables the router to send all traps to the host myhost.cisco.com using the community string public:

```
snmp-server enable traps
snmp-server host myhost.cisco.com public
```

The following example will not send traps to any host. The BGP traps are enabled for all hosts, but only the ISDN traps are enabled to be sent to a host.

```
snmp-server enable traps bgp
snmp-server host bob public isdn
```

The following example enables the router to send all inform requests to the host myhost.cisco.com using the community string public:

```
snmp-server enable traps
snmp-server host myhost.cisco.com informs version 2c public
```

### SNMP Manager Example

The following example enables the SNMP manager and sets the session timeout to a larger value than the default:

```
snmp-server manager
snmp-server manager session-timeout 1000
```

## RMON Alarm and Event Examples

The following example enables the **rmon event** command:

```
rmon event 1 log trap eventtrap description "High ifOutErrors" owner sdurham
```

This example creates RMON event number 1, which is defined as *High ifOutErrors*, and generates a log entry when the event is triggered by an alarm. The user *sdurham* owns the row that is created in the event table by this command. This example also generates a Simple Network Management Protocol (SNMP) trap when the event is triggered.

The following example configures an RMON alarm using the **rmon alarm** command:

```
rmon alarm 10 ifEntry.20.1 20 delta rising-threshold 15 1 falling-threshold 0 owner
jjohnson
```

This example configures RMON alarm number 10. The alarm monitors the MIB variable *ifEntry.20.1* once every 20 seconds until the alarm is disabled, and checks the change in the variable's rise or fall. If the *ifEntry.20.1* value shows a MIB counter increase of 15 or more, such as

from 100000 to 100015, the alarm is triggered. The alarm in turn triggers event number 1, which is configured with the **rmon event** command. Possible events include a log entry or an SNMP trap. If the *ifEntry.20.1* value changes by 0, the alarm is reset and can be triggered again.

## Response Time Reporter Examples

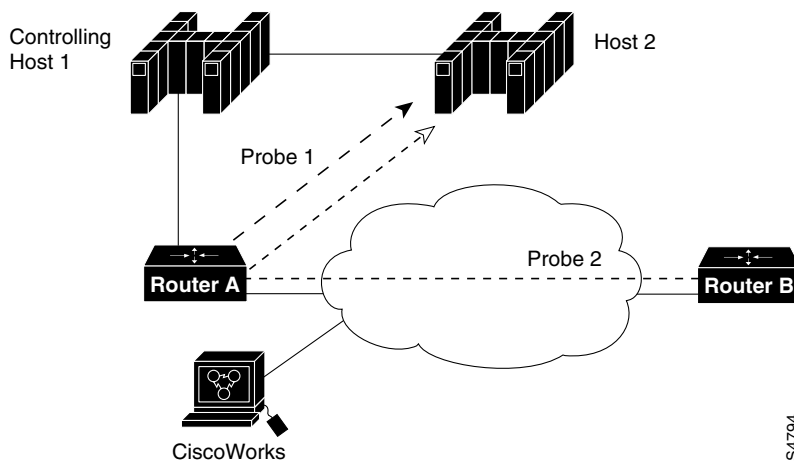
The following sections contain examples of setting up probes on the router to monitor network performance and send notifications:

- Perform Normative Analysis for SNA LU2
- Perform Troubleshooting for IP/ICMP
- Configure a Trigger for Connection Loss

### Perform Normative Analysis for SNA LU2

In the example shown in Figure 362, probe 1 is configured from router A to host 2, and probe 2 is configured from router B to host 2 to perform a normative analysis of the network to determine a baseline from which triggers (and reactions in general) are then configured. Also, two SNA Physical Units (PUs) are assumed to be configured: CWBC0A and CWBC0B. For information on configuring PUs, see the **dspu host** or the **sna host** command in the *Bridging and IBM Networking Command Reference*.

**Figure 362** Configure Probes for Normative Analysis—SNA LU2



### Router A's Configuration:

```
RouterA(config)# rtr 1
RouterA(config-rtr)# type echo protocol snaLU2EchoAppl CWBC0A
RouterA(config-rtr)# exit
RouterA(config)# rtr schedule 1 start-time now
RouterA(config)# exit
```

### Router B's Configuration:

```
RouterB(config)# rtr 2
RouterB(config-rtr)# type echo protocol snaLU2EchoAppl CWBC0B
RouterB(config-rtr)# exit
RouterB(config)# rtr schedule 1 start-time now
RouterB(config)# exit
```

### Configuration Files for Router A and Router B

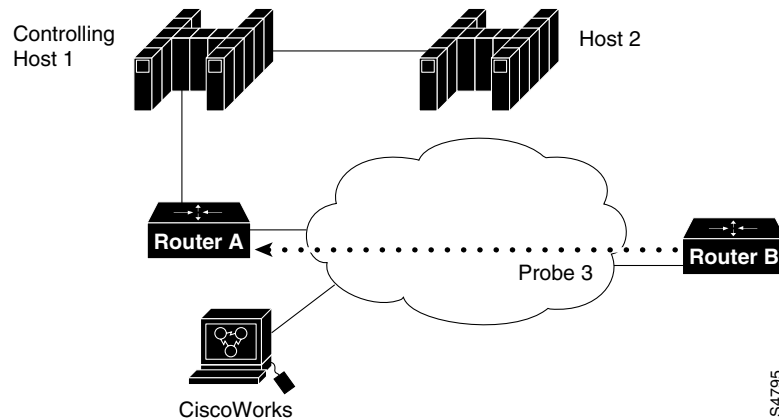
After you save the configurations (using the **copy running-config startup-config** command), the following information is stored in the configuration files. Note the addition of the “kept” commands in the configuration file. They are automatically included because they differ depending on the **type** you specify for the probe.

```
!Router A Configuration File
! Router A's PU Configuration
sna host CWBC0A xid-snd 05dcc00a rmac 4001.3745.1088 rsap 4 lsap 12 focalpoint
rtr 1
  type echo protocol snaLU2EchoAppl CWBC0A
  paths-of-statistics-kept 1
  hops-of-statistics-kept 1
  samples-of-history-kept 1
rtr schedule 1 start-time now

!Router B Configuration File
!Router B's PU Configuration from the Configuration File:
sna host CWBC0B xid-snd 05dcc00b rmac 4001.3745.1088 rsap 4 lsap 12 focalpoint
rtr 2
  type echo protocol snaLU2EchoAppl CWBC0B
  paths-of-statistics-kept 1
  hops-of-statistics-kept 1
  samples-of-history-kept 1
rtr schedule 2 start-time now
```

### Perform Troubleshooting for IP/ICMP

In the example shown in Figure 363, probe 3 is configured from router B to router A to perform troubleshooting of the network to determine a network problem from which triggers (and reactions in general) are then configured.

**Figure 363 Configure a Probe for Troubleshooting—IP/ICMP**

This example sets up a **pathEcho** (with history) pending entry from router B to router A via IP/ICMP. It will attempt to execute three times in 25 seconds (first attempt starts at 0 seconds) and will keep those three times with three buckets. It can be started five times before wrapping over stored history (lives 5). Because this configuration keeps history, it uses more RAM on the router.

#### Router B's Configuration:

```
RouterB(config)# rtr 3
RouterB(config-rtr)# type pathEcho protocol ipIcmpEcho RouterA
RouterB(config-rtr)# frequency 10
RouterB(config-rtr)# lives-of-history-kept 5
RouterB(config-rtr)# buckets-of-history-kept 3
RouterB(config-rtr)# filter-for-history all
RouterB(config-rtr)# exit
RouterB(config)# rtr schedule 3 life 25
RouterB(config)# exit
```

#### Configuration File for Router B

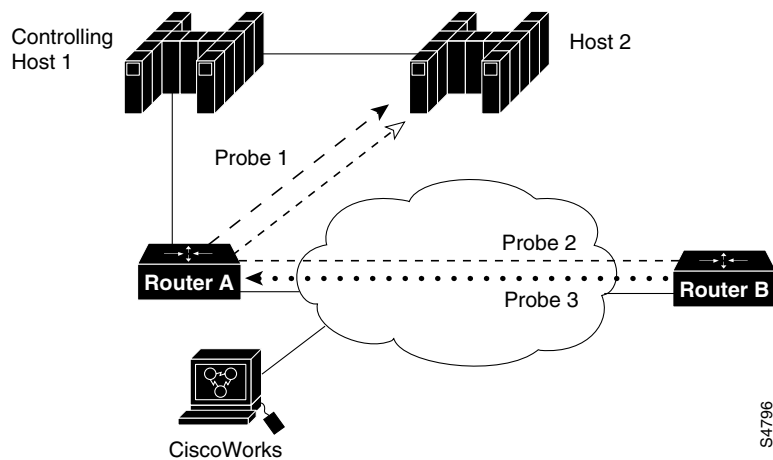
After you save the configuration (using the **copy running-config startup-config** command), the following information is stored in the configuration file. Note the addition of commands in the configuration file. They are automatically included because they differ depending on the **type** you specify for the probe.

```
rtr 3
type pathEcho protocol ipIcmpEcho 172.28.161.21
frequency 10
response-data-size 1
lives-of-history-kept 5
buckets-of-history-kept 3
filter-for-history all
rtr schedule 3 life 25 start-time pending
```

#### Configure a Trigger for Connection Loss

Figure 364 shows probes 1, 2, and 3 in the network. This example shows how to configure a trigger if probe 2 encounters a connection loss from router B to host 2. If a connection loss occurs between router B and host 2, a trap is issued, an SNA NMVT Alert is issued, and probe 3's state is changed to "active."

**Figure 364** Configure a Trigger for Connection Loss



**Router B's Configuration:**

```
RouterB(config)# rtr reaction-configuration 2 connection-loss-enable  
                  action-type trapNmvtAndTrigger  
RouterB(config)# rtr reaction-trigger 2 3
```

---

**Note** The probe numbers need only be unique within one router. The examples shown use three different probe numbers for clarity.

---