CHAPTER 8

Checking Port Status and Connectivity

This chapter describes how to check switch port status and connectivity on the Catalyst 4500 series switch.

This chapter includes the following major sections:

- Checking Module Status, page 8-2
- Checking Interfaces Status, page 8-2
- Displaying MAC Addresses, page 8-3
- Checking Cable Status Using Time Domain Reflectometer, page 8-3
- Using Telnet, page 8-5
- Changing the Logout Timer, page 8-6
- Monitoring User Sessions, page 8-6
- Using Ping, page 8-7
- Using IP Traceroute, page 8-8
- Using Layer 2 Traceroute, page 8-10
- Configuring ICMP, page 8-12

Note For complete syntax and usage information for the switch commands used in this chapter, look at the *Cisco Catalyst 4500 Series Switch Command Reference* and related publications at this location:

http://www.cisco.com/en/US/products/hw/switches/ps4324/index.html

If the command is not found in the Catalyst 4500 Command Reference, it is located in the larger Cisco IOS library. Refer to the *Catalyst 4500 Series Switch Cisco IOS Command Reference* and related publications at this location:

http://www.cisco.com/en/US/products/ps6350/index.html

Checking Module Status

The Catalyst 4500 series switch is a multimodule system. You can see which modules are installed, as well as the MAC address ranges and version numbers for each module, by using the **show module** command. You can use the [*mod_num*] argument to specify a particular module number to see detailed information on that module.

This example shows how to check module status for all modules on your switch:

```
Switch# show module all

Mod  Ports Card Type                              Model              Serial No.
---+-----+--------------------------------------+------------------+-----------
 1     2   1000BaseX (GBIC) Supervisor Module     WS-X4014          JAB012345AB
 5    24   10/100/1000BaseTX (RJ45)               WS-X4424-GB-RJ45 JAB045304EY
 6    48   10/100BaseTX (RJ45)                    WS-X4148          JAB023402QK

 M MAC addresses                   Hw  Fw                Sw               Stat
--+--------------------------------+---+-----------------+---------------+-----
 1 0004.dd46.9f00 to 0004.dd46.a2ff 0.0 12.1(10r)EW(1.21) 12.1(10)EW(1)    Ok
 5 0050.3e7e.1d70 to 0050.3e7e.1d87 0.0                                    Ok
 6 0050.0f10.2370 to 0050.0f10.239f 1.0                                    Ok
Switch#
```

Checking Interfaces Status

You can view summary or detailed information on the switch ports using the **show interfaces status** command. To see summary information on all ports on the switch, enter the **show interfaces status** command with no arguments. Specify a particular module number to see information on the ports on that module only. Enter both the module number and the port number to see detailed information about the specified port.

To apply configuration commands to a particular port, you must specify the appropriate logical module. For more information, see the "Checking Module Status" section on page 8-2.

This example shows how to display the status of all interfaces on a Catalyst 4500 series switch, including transceivers. Output of this command displays "Unapproved GBIC" for non-Cisco transceivers:

```
Switch# show interfaces status

Port     Name               Status       Vlan       Duplex  Speed Type
Gi1/1                       notconnect   1            auto   auto No Gbic
Gi1/2                       notconnect   1            auto   auto No Gbic
Gi5/1                       notconnect   1            auto   auto 10/100/1000-TX
Gi5/2                       notconnect   1            auto   auto 10/100/1000-TX
Gi5/3                       notconnect   1            auto   auto 10/100/1000-TX
Gi5/4                       notconnect   1            auto   auto 10/100/1000-TX
Fa6/1                       connected    1          a-full  a-100 10/100BaseTX
Fa6/2                       connected    2          a-full  a-100 10/100BaseTX
Fa6/3                       notconnect   1            auto   auto 10/100BaseTX
Fa6/4                       notconnect   1            auto   auto 10/100BaseTX

Switch#
```

This example shows how to display the status of interfaces in error-disabled state:

```
Switch# show interfaces status err-disabled
Port    Name               Status         Reason
Fa9/4                      err-disabled   link-flap
informational error message when the timer expires on a cause
-----------------------------------------------------------------
5d04h:%PM-SP-4-ERR_RECOVER:Attempting to recover from link-flap err-disable state on Fa9/4
Switch#
```

Displaying MAC Addresses

In addition to displaying the MAC address range for a module using the **show module** command, you can display the MAC address table information of a specific MAC address or a specific interface in the switch using the **show mac-address-table address** and **show mac-address-table interface** commands.

This example shows how to display MAC address table information for a specific MAC address:

```
Switch# show mac-address-table address 0050.3e8d.6400
vlan   mac address     type    protocol  qos             ports
-----+---------------+--------+---------+---+--------------------------------
 200  0050.3e8d.6400  static   assigned  --  Switch
 100  0050.3e8d.6400  static   assigned  --  Switch
   5  0050.3e8d.6400  static   assigned  --  Switch
   4  0050.3e8d.6400  static        ipx  --  Switch
   1  0050.3e8d.6400  static        ipx  --  Switch
   1  0050.3e8d.6400  static   assigned  --  Switch
   4  0050.3e8d.6400  static   assigned  --  Switch
   5  0050.3e8d.6400  static        ipx  --  Switch
 100  0050.3e8d.6400  static        ipx  --  Switch
 200  0050.3e8d.6400  static        ipx  --  Switch
 100  0050.3e8d.6400  static      other  --  Switch
 200  0050.3e8d.6400  static      other  --  Switch
   5  0050.3e8d.6400  static      other  --  Switch
   4  0050.3e8d.6400  static         ip  --  Switch
   1  0050.3e8d.6400  static         ip  --  Route
   1  0050.3e8d.6400  static      other  --  Switch
   4  0050.3e8d.6400  static      other  --  Switch
   5  0050.3e8d.6400  static         ip  --  Switch
 200  0050.3e8d.6400  static         ip  --  Switch
 100  0050.3e8d.6400  static         ip  --  Switch
Switch#
```

This example shows how to display MAC address table information for a specific interface:

```
Switch# show mac-address-table interface gigabit 1/1
Multicast Entries
 vlan    mac address     type    ports
-------+---------------+-------+-----------------------------------------
   1    ffff.ffff.ffff system Switch,Gi6/1,Gi6/2,Gi6/9,Gi1/1
Switch#
```

Checking Cable Status Using Time Domain Reflectometer

The Time Domain Reflectometer (TDR) feature allows you to determine if cable is OPEN or SHORT when it is at fault.

Overview

With TDR, you can check the status of copper cables on the 48-port 10/100/1000 BASE-T modules for the Catalyst 4500 series switch. TDR detects a cable fault by sending a signal through the cable and reading the signal that is reflected back. All or part of the signal can be reflected back either by cable defects or by the end of the cable.

Note Four pairs of standard category 5 cable exist. Each pair can assume one of the following states: open (not connected), broken, shorted, or terminated. The TDR test detects all four states and displays the first three as "Fault" conditions, and displays the fourth as "Terminated." Although the CLI output is shown, the cable length is displayed only if the state is "Faulty."

TDR feature is supported on the following modules:

WS-X4524-GB-RJ45V

WS-X4548-GB-RJ45

WS-X4548-GB-RJ45V

WS-X4548-GB-RJ45V+

WS-X4548-RJ45V+

WS-X4748-RJ45+E

WS-X4748-RJ45V+E

WS-C4948E

TDR detects a cable fault by sending a signal along its wires and depending on the reflected signal it can determine roughly where a cable fault could be. The variations on how TDR signal is reflected back determine the results on TDR. On cat4k products, we only support cable fault types: OPEN or SHORT. We do display Terminated status in case cable is proper terminated and this is done for illustrative purpose.

Running the TDR Test

To start the TDR test, perform this task in privileged mode:

	Command	Purpose
Step 1	`Switch# test cable-diagnostics tdr {interface {interface interface-number}}`	Starts the TDR test.
Step 2	`Switch# show cable-diagnostics tdr {interface {interface interface-number}}`	Displays the TDR test counter information.

This example shows how to start the TDR test on port 1 on module 2:

```
Switch# test cable-diagnostics tdr int gi2/1
Switch#
```

This example shows the message that displays when the TDR test is not supported on a module:

```
Switch# test cable-diagnostics tdr int gi2/1
00:03:15:%C4K_IOSDIAGMAN-4-TESTNOTSUPPORTEDONMODULE: Online cable
diag tdr test is not supported on this module
Switch#
```

This example shows how to display TDR test results for a port:

```
Switch# show cable-diagnostics tdr interface gi4/13
Interface Speed  Local pair Cable length Remote channel Status
Gi4/13    0Mbps   1-2        102 +-2m     Unknown       Fault
                  3-6        100 +-2m     Unknown       Fault
                  4-5        102 +-2m     Unknown       Fault
                  7-8        102 +-2m     Unknown       Fault
```

Note After this command is deprecated, use the diagnostic start and the **show diagnostic result** commands to run the TDR test and display the test results.

Note TDR is a port test; the port cannot handle traffic for the duration of the test (generally, 1 minute).

TDR Guidelines

The following guidelines apply to the use of TDR:

- If you connect a port undergoing a TDR test to an Auto-MDIX enabled port, the TDR result might be invalid. In those instances, the port on the WS-X4148-RJ45V should be administratively down before the start of the TDR test.
- If you connect a port undergoing a TDR test to a 100BASE-T port such as that on the WS-X4148-RJ45V, the unused pairs (4-5 and 7-8) will be reported as faulty because the remote end does not terminate these pairs.
- Do not change the port configuration while the TDR test is running.
- Due to cable characteristics, you should run the TDR test multiple times to get accurate results.
- Do not change port status (for example, remove the cable at the near or far end) because the results might be inaccurate.
- TDR works best if the test cable is disconnected from the remote port. Otherwise, it might be difficult for you to interpret results correctly.
- TDR operates across four wires. Depending on the cable conditions, the status might show one pair is OPEN or SHORT while all other wire pairs display as faulty. This operation is acceptable because you should declare a cable faulty provided one pair of wires is found to be OPEN or SHORT.
- TDR determines how poor a cable is functioning rather than where a faulty cable is located.
- When TDR locates a faulty cable, you should still use an offline cable diagnosis tool to better diagnose the problem.
- TDR results might differ between runs on different Catalyst 4500 modules because of resolution difference in TDR implementations. When this occurs, you should refer to offline cable diagnosis tool.

Using Telnet

You can access the switch command-line interface (CLI) using Telnet. In addition, you can use Telnet from the switch to access other devices in the network. You can have up to eight simultaneous Telnet sessions.

Before you can open a Telnet session to the switch, you must first set the IP address (and in some cases the default gateway) for the switch. For information about setting the IP address and default gateway, see Chapter 3, "Configuring the Switch for the First Time."

Note To establish a Telnet connection to a host by using the hostname, configure and enable DNS.

To establish a Telnet connection to another device on the network from the switch, perform this task:

Command	Purpose
`Switch# telnet` *host* `[`*port*`]`	Opens a Telnet session to a remote host.

This example shows how to establish a Telnet connection from the switch to the remote host named labsparc:

```
Switch# telnet labsparc
Trying 172.16.10.3...
Connected to labsparc.
Escape character is '^]'.

UNIX(r) System V Release 4.0 (labsparc)

login:
```

Changing the Logout Timer

The logout timer automatically disconnects a user from the switch when the user is idle for longer than the specified time. To set the logout timer, perform this task:

Command	Purpose
`Switch# logoutwarning` *number*	Changes the logout timer value (a timeout value of 0 prevents idle sessions from being disconnected automatically). Use the **no** keyword to return to the default value.

Monitoring User Sessions

You can display the currently active user sessions on the switch using the **show users** command. The command output lists all active console port and Telnet sessions on the switch.

To display the active user sessions on the switch, perform this task in privileged EXEC mode:

Command	Purpose
`Switch# show users [all]`	Displays the currently active user sessions on the switch.

This example shows the output of the **show users** command when local authentication is enabled for console and Telnet sessions (the asterisk [*] indicates the current session):

```
Switch# show users
    Line       User       Host(s)              Idle       Location
*  0 con 0                idle                 00:00:00

  Interface      User        Mode                    Idle     Peer Address

Switch# show users all
    Line       User       Host(s)              Idle       Location
*  0 con 0                idle                 00:00:00
   1 vty 0                                     00:00:00
   2 vty 1                                     00:00:00
   3 vty 2                                     00:00:00
   4 vty 3                                     00:00:00
   5 vty 4                                     00:00:00

  Interface      User        Mode                    Idle     Peer Address
Switch#
```

To disconnect an active user session, perform this task:

Command	Purpose
`Switch# disconnect {console | ip_addr}`	Disconnects an active user session on the switch.

This example shows how to disconnect an active console port session and an active Telnet session:

```
Switch> disconnect console
Console session disconnected.
Console> (enable) disconnect tim-nt.bigcorp.com
Telnet session from tim-nt.bigcorp.com disconnected. (1)
Switch# show users
  Session  User             Location
  -------- ---------------- -------------------------
  telnet   jake             jake-mac.bigcorp.com
* telnet   suzy             suzy-pc.bigcorp.com
Switch#
```

Using Ping

These sections describe how to use IP ping:

- Understanding How Ping Works, page 8-7
- Running Ping, page 8-8

Understanding How Ping Works

You can use the **ping** command to verify connectivity to remote hosts. If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or configure a router to route between those subnets.

The **ping** command is configurable from normal executive and privileged EXEC mode. Ping returns one of the following responses:

- Normal response—The normal response (*hostname* is alive) occurs in 1 to 10 seconds, depending on network traffic.
- Destination does not respond—If the host does not respond, a No Answer message is returned.
- Unknown host—If the host does not exist, an Unknown Host message is returned.
- Destination unreachable—If the default gateway cannot reach the specified network, a Destination Unreachable message is returned.
- Network or host unreachable—If there is no entry in the route table for the host or network, a Network or Host Unreachable message is returned.

To stop a ping in progress, press **Ctrl-C**.

Running Ping

To ping another device on the network from the switch, perform this task in normal executive and privileged EXEC mode:

Command	Purpose
`Switch# ping` *host*	Checks connectivity to a remote host.

This example shows how to ping a remote host from normal executive mode:

```
Switch# ping labsparc
labsparc is alive
Switch> ping 72.16.10.3
12.16.10.3 is alive
Switch#
```

This example shows how to enter a **ping** command in privileged EXEC mode specifying the number of packets, the packet size, and the timeout period:

```
Switch# ping
Target IP Address []: 12.20.5.19
Number of Packets [5]: 10
Datagram Size [56]: 100
Timeout in seconds [2]: 10
Source IP Address [12.20.2.18]: 12.20.2.18
!!!!!!!!!!

----12.20.2.19 PING Statistics----
10 packets transmitted, 10 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 1/1/1
Switch
```

Using IP Traceroute

These sections describe how to use IP traceroute feature:

- Understanding How IP Traceroute Works, page 8-9
- Running IP Traceroute, page 8-9

Understanding How IP Traceroute Works

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Layer 2 switches can participate as the source or destination of the **trace** command but will not appear as a hop in the **trace** command output.

The **trace** command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends back an Internet Control Message Protocol (ICMP) Time-Exceeded message to the sender. Traceroute determines the address of the first hop by examining the source address field of the ICMP Time-Exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the Time-Exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host or until the maximum TTL is reached.

To determine when a datagram reaches its destination, traceroute sets the UDP destination port in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram with an unrecognized port number, it sends an ICMP Port Unreachable error message to the source. The Port Unreachable error message indicates to traceroute that the destination has been reached.

Running IP Traceroute

To trace the path that packets take through the network, perform this task in EXEC or privileged EXEC mode:

Command	Purpose
Switch# **trace** [*protocol*] [*destination*]	Runs IP traceroute to trace the path that packets take through the network.

This example shows use the **trace** command to display the route a packet takes through the network to reach its destination:

```
Switch# trace ip ABA.NYC.mil

Type escape sequence to abort.
Tracing the route to ABA.NYC.mil (26.0.0.73)
  1 DEBRIS.CISCO.COM (192.180.1.6) 1000 msec 8 msec 4 msec
  2 BARRNET-GW.CISCO.COM (192.180.16.2) 8 msec 8 msec 8 msec
  3 EXTERNAL-A-GATEWAY.STANFORD.EDU (192.42.110.225) 8 msec 4 msec 4 msec
  4 BB2.SU.BARRNET.NET (192.200.254.6) 8 msec 8 msec 8 msec
  5 SU.ARC.BARRNET.NET (192.200.3.8) 12 msec 12 msec 8 msec
  6 MOFFETT-FLD-MB.in.MIL (192.52.195.1) 216 msec 120 msec 132 msec
  7 ABA.NYC.mil (26.0.0.73) 412 msec 628 msec 664 msec
Switch#
```

Using Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. It determines the path by using the MAC address tables of the switches in the path. When the switch detects a device in the path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

If you want the switch to trace the path from a host on a source device to a host on a destination device, the switch can identify only the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

These sections describe how to use the Layer 2 traceroute feature:

- Layer 2 Traceroute Usage Guidelines, page 8-10
- Running Layer 2 Traceroute, page 8-11

Layer 2 Traceroute Usage Guidelines

These are the Layer 2 traceroute usage guidelines:

- CDP must be enabled on all the devices in the network. For Layer 2 traceroute to functional properly, do not disable CDP.

  If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices.

  Note For more information about enabling CDP, see Chapter 20, "Configuring CDP."

- All switches in the physical path must have IP connectivity. When a switch is reachable from another switch, you can test connectivity by using the **ping** command in privileged EXEC mode.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** command in privileged EXEC mode on a switch that is not in the physical path from the source device to the destination device. All switches in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.

- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the switch uses Address Resolution Protocol (ARP) to associate the IP address with the corresponding MAC address and the VLAN ID.
  - If an ARP entry exists for the specified IP address, the switch uses the associated MAC address and identifies the physical path.
  - If an ARP entry does not exist, the switch sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- This feature is not supported in Token Ring VLANs.

Running Layer 2 Traceroute

To display the physical path that a packet takes from a source device to a destination device, perform either one of these tasks in privileged EXEC mode:

Command	Purpose
Switch# **traceroute mac** {*source-mac-address*} {*destination-mac-address*}	Runs Layer 2 traceroute to trace the path that packets take through the network.

or

Command	Purpose
Switch# **traceroute mac ip** {*source-mac-address*} {*destination-mac-address*}	Runs IP traceroute to trace the path that packets take through the network.

These examples show how to use the **traceroute mac** and **traceroute mac ip** commands to display the physical path a packet takes through the network to reach its destination:

```
Switch# traceroute mac 0000.0201.0601 0000.0201.0201

Source 0000.0201.0601 found on con6[WS-C2950G-24-EI] (2.2.6.6)
con6 (2.2.6.6) :Fa0/1 => Fa0/3
con5                 (2.2.5.5        )  :    Fa0/3 => Gi0/1
con1                 (2.2.1.1        )  :    Gi0/1 => Gi0/2
con2                 (2.2.2.2        )  :    Gi0/2 => Fa0/1
Destination 0000.0201.0201 found on con2[WS-C3550-24] (2.2.2.2)
Layer 2 trace completed
Switch#

Switch# traceroute mac ip 2.2.66.66 2.2.22.22 detail
Translating IP to mac .....
2.2.66.66 => 0000.0201.0601
2.2.22.22 => 0000.0201.0201
```

```
Source 0000.0201.0601 found on con6[WS-C2950G-24-EI] (2.2.6.6)
con6 / WS-C2950G-24-EI / 2.2.6.6 :
        Fa0/1 [auto, auto] => Fa0/3 [auto, auto]
con5 / WS-C2950G-24-EI / 2.2.5.5 :
        Fa0/3 [auto, auto] => Gi0/1 [auto, auto]
con1 / WS-C3550-12G / 2.2.1.1 :
        Gi0/1 [auto, auto] => Gi0/2 [auto, auto]
con2 / WS-C3550-24 / 2.2.2.2 :
        Gi0/2 [auto, auto] => Fa0/1 [auto, auto]
Destination 0000.0201.0201 found on con2[WS-C3550-24] (2.2.2.2)
Layer 2 trace completed.
Switch#
```

Configuring ICMP

Internet Control Message Protocol (ICMP) provides many services that control and manage IP connections. ICMP messages are sent by routers or access servers to hosts or other routers when a problem is discovered with the Internet header. For detailed information on ICMP, refer to RFC 792.

Enabling ICMP Protocol Unreachable Messages

If the Cisco IOS software receives a nonbroadcast packet that uses an unknown protocol, it sends an ICMP Protocol Unreachable message back to the source.

Similarly, if the software receives a packet that it is unable to deliver to the ultimate destination because it knows of no route to the destination address, it sends an ICMP Host Unreachable message to the source. This feature is enabled by default.

To enable the generation of ICMP Protocol Unreachable and Host Unreachable messages, enter the following command in interface configuration mode:

Command	Purpose
`Switch (config-if)# [no] ip unreachables`	Enables ICMP destination unreachable messages. Use the **no** keyword to disable the ICMP destination unreachable messages.

Caution If you enter the **no ip unreachables** command, you will break the path MTU discovery" functionality. Routers in the middle of the network might be forced to fragment packets.

To limit the rate that Internet Control Message Protocol (ICMP) destination unreachable messages are generated, perform this task:

Command	Purpose
`Switch (config)# [no] ip icmp rate-limit unreachable [df] milliseconds`	Limits the rate that ICMP destination messages are generated. Use the **no** keyword to remove the rate limit and reduce the CPU usage.

Enabling ICMP Redirect Messages

Data routes are sometimes less than optimal. For example, it is possible for the router to be forced to resend a packet through the same interface on which it was received. If this occurs, the Cisco IOS software sends an ICMP Redirect message to the originator of the packet telling the originator that the router is on a subnet directly connected to the receiving device, and that it must forward the packet to another system on the same subnet. The software sends an ICMP Redirect message to the packet's originator because the originating host presumably could have sent that packet to the next hop without involving this device at all. The Redirect message instructs the sender to remove the receiving device from the route and substitute a specified device representing a more direct path. This feature is enabled by default.

However, when Hot Standby Router Protocol (HSRP) is configured on an interface, ICMP Redirect messages are disabled (by default) for the interface. For more information on HSRP, refer to the following URL:

http://www.cisco.com/en/US/docs/ios/ipapp/configuration/guide/ipapp_hsrp_ps6350_TSD_Products_Configuration_Guide_Chapter.html

To enable the sending of ICMP Redirect messages if the Cisco IOS software is forced to resend a packet through the same interface on which it was received, enter the following command in interface configuration mode:

Command	Purpose
`Switch (config-if)# [no] ip redirects`	Enables ICMP Redirect messages. Use the **no** keyword to disable the ICMP Redirect messages and reduce CPU usage.

Enabling ICMP Mask Reply Messages

Occasionally, network devices must know the subnet mask for a particular subnetwork in the internetwork. To obtain this information, devices can send ICMP Mask Request messages. These messages are responded to by ICMP Mask Reply messages from devices that have the requested information. The Cisco IOS software can respond to ICMP Mask Request messages if the ICMP Mask Reply function is enabled.

To have the Cisco IOS software respond to ICMP mask requests by sending ICMP Mask Reply messages, perform this task:

Command	Purpose
`Switch (config-if)# [no] ip mask-reply`	Enables response to ICMP destination mask requests. Use the **no** keyword to disable this functionality.