



Troubleshooting

This chapter describes how to identify and resolve Catalyst 3550 software problems related to the Cisco IOS software. Depending on the nature of the problem, you can use the command-line interface (CLI), the device manager, or Network Assistant to identify and solve problems.



Note

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the *Cisco IOS Command Summary for Cisco IOS Release 12.1*.

This chapter consists of these sections:

- [Using Recovery Procedures, page 36-1](#)
- [Preventing Autonegotiation Mismatches, page 36-10](#)
- [GBIC Module Security and Identification, page 36-11](#)
- [Diagnosing Connectivity Problems, page 36-11](#)
- [Troubleshooting Power over Ethernet Switch Ports, page 36-16](#)
- [Using Debug Commands, page 36-17](#)
- [Using the show forward Command, page 36-19](#)
- [Using the crashinfo File, page 36-21](#)



Note

If after applying ACLs, you are experiencing packet performance problems or receiving messages about TCAM capacity, see the [“Displaying ACL Resource Usage and Configuration Problems”](#) section on [page 27-42](#).

Using Recovery Procedures

These recovery procedures require that you have physical access to the switch:

- [Recovering from Corrupted Software, page 36-2](#)
- [Recovering from a Lost or Forgotten Password, page 36-3](#)
- [Recovering from a Command Switch Failure, page 36-6](#)
- [Recovering from Lost Member Connectivity, page 36-10](#)

Recovering from Corrupted Software

Switch software can be corrupted during an upgrade, by downloading the wrong file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

This procedure uses the Xmodem protocol to recover from a corrupt or wrong image file. There are many software packages that support the Xmodem protocol, and this procedure is largely dependent on the emulation software you are using.

Follow these steps to recover from corrupted software:

Step 1 Connect a PC with terminal-emulation software supporting the XMODEM protocol to the switch console port.

Step 2 Set the line speed on the emulation software to 9600 baud.

Step 3 Unplug the switch power cord.

Step 4 Press the **Mode** button, and at the same time, reconnect the power cord to the switch.

You can release the **Mode** button a second or two after the LED above port 1X turns off. Several lines of information about the software appear along with instructions:

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system, and finish loading the operating system
software#
```

```
flash_init
load_helper
boot
```

Step 5 Initialize the flash file system:

```
switch# flash_init
```

Step 6 If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

Step 7 Load any helper files:

```
switch# load_helper
```

Step 8 Start the file transfer by using the XMODEM protocol.

```
switch# copy xmodem: flash:image_filename.bin
```

Step 9 After the XMODEM request appears, use the appropriate command on the terminal-emulation software to start the transfer and to copy the software image into flash memory.

Recovering from a Lost or Forgotten Password

An end user with physical access to the switch can recover from a lost password by interrupting the boot process during power-on and by entering a new password. This is the default configuration for Catalyst 3550 switches.

**Note**

On Catalyst 3550 Fast Ethernet switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password on a Catalyst 3550 Fast Ethernet switch and password recover has been disabled, a status message shows this during the recovery process.

Follow these steps if you have forgotten or lost the switch password.

- Step 1** Connect a terminal or PC with terminal-emulation software to the switch console port.
- Step 2** Set the line speed on the emulation software to 9600 baud.
- Step 3** Unplug the switch power cord.
- Step 4** Reconnect the power cord to the switch and, within 15 seconds, press the **Mode** button while the System LED above port 1X is still flashing green. Continue pressing the **Mode** button until the System LED turns briefly amber and then solid green; then release the **Mode** button.

Several lines of information about the software appear with instructions, informing you if the password recovery procedure has been disabled or not.

- If you see a message that begins with this:

```
The system has been interrupted prior to initializing the flash file system. The
following commands will initialize the flash file system
```

go to the [“Password Recovery with Password Recovery Enabled”](#) section on page 36-3, and follow the steps.

- If you see a message that begins with this:

```
The password-recovery mechanism has been triggered, but is currently disabled.
```

go to the [“Procedure with Password Recovery Disabled”](#) section on page 36-5, and follow the steps.

Password Recovery with Password Recovery Enabled

If the password-recovery mechanism is enabled, this message appears:

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system, and finish loading the operating system
software:
```

```
flash_init
load_helper
boot
```

Follow these steps when the password-recovery is enabled:

Step 1 Initialize the flash file system:

```
switch# flash_init
```

Step 2 If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

Step 3 Load any helper files:

```
switch# load_helper
```

Step 4 Display the contents of flash memory:

```
switch# dir flash:
```

The switch file system appears in the directory.

Step 5 Rename the configuration file to config.text.old.

This file contains the password definition.

```
switch# rename flash:config.text flash:config.text.old
```

Step 6 Boot the system:

```
switch# boot
```

You are prompted to start the setup program. Enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 7 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 8 Rename the configuration file to its original name:

```
Switch# rename flash:config.text.old flash:config.text
```

Step 9 Copy the configuration file into memory:

```
Switch# copy flash:config.text system:running-config  
Source filename [config.text]?  
Destination filename [running-config]?
```

Press **Return** in response to the confirmation prompts.

The configuration file is now reloaded, and you can change the password.

Step 10 Enter global configuration mode:

```
Switch# configure terminal
```

Step 11 Change the password:

```
Switch (config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 12 Return to privileged EXEC mode:

```
Switch (config)# exit  
Switch#
```

Step 13 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.



Note

This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To re-enable the interface, enter the **interface vlan vlan-id** global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled. Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point. However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.
```

```
Would you like to reset the system back to the default configuration (y/n)?
```



Caution

Returning the switch to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup switch and VLAN configuration files.

- If you enter **n** (no), the normal boot process continues as if the **Mode** button had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

```
Press Enter to continue.....
```
- If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.

Follow these steps when the password-recovery mechanism is disabled:

Step 1 Elect to continue with password recovery and lose the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? y
```

Step 2 Load any helper files:

```
Switch# load_helper
```

Step 3 Display the contents of flash memory:

```
switch# dir flash:
```

The switch file system appears in the directory.

Step 4 Boot the system:

```
Switch# boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 5 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 6 Enter global configuration mode:

```
Switch# configure terminal
```

Step 7 Change the password:

```
Switch (config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 8 Return to privileged EXEC mode:

```
Switch (config)# exit
Switch#
```

Step 9 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

**Note**

This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To re-enable the interface, enter the **interface vlan *vlan-id*** global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Step 10 You must now reconfigure the switch. If the system administrator has the backup switch and VLAN configuration files available, you should use those.

Recovering from a Command Switch Failure

This section describes how to recover from a failed command switch. You can configure a redundant command switch group by using the Hot Standby Router Protocol (HSRP). For more information, see [Chapter 5, “Clustering Switches”](#), [Chapter 31, “Configuring HSRP”](#), and the *Getting Started with Cisco Network Assistant*, available on Cisco.com.

**Note**

HSRP is the preferred method for supplying redundancy to a cluster.

If you have not configured a standby command switch, and your command switch loses power or fails in some other way, management contact with the member switches is lost, and you must install a new command switch. However, connectivity between switches that are still connected is not affected, and the member switches forward packets as usual. You can manage the members as standalone switches through the console port or, if they have IP addresses, through the other management interfaces.

You can prepare for a command switch failure by assigning an IP address to a member switch or another switch that is command-capable, making a note of the command-switch password, and cabling your cluster to have redundant connectivity between the member switches and the replacement command switch. This section describes two solutions for replacing a failed command switch:

- Replacing a failed command switch with a cluster member
- Replacing a failed command switch with another switch

For information on command-capable switches, see the release notes.

Replacing a Failed Command Switch with a Cluster Member

To replace a failed command switch with a command-capable member in the same cluster, follow these steps:

-
- Step 1** Disconnect the command switch from the member switches, and physically remove it from the cluster.
- Step 2** Insert the member switch in place of the failed command switch, and duplicate its connections to the cluster members.
- Step 3** Start a CLI session on the new command switch.
- You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see the switch hardware installation guide.
- Step 4** At the switch prompt, enter privileged EXEC mode:
- ```
Switch> enable
Switch#
```
- Step 5** Enter the password of the *failed command switch*.
- Step 6** Enter global configuration mode.
- ```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```
- Step 7** Remove the member switch from the cluster.
- ```
Switch(config)# no cluster commander-address
```
- Step 8** Return to privileged EXEC mode.
- ```
Switch(config)# end
Switch#
```

- Step 9** Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter **setup**, and press **Return**.

```
Switch# setup
    --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.

Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system

Would you like to enter basic management setup? [yes/no]:
```

- Step 10** Enter **Y** at the first prompt.

The prompts in the setup program vary depending on the member switch you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
or
```

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

- Step 11** Respond to the questions in the setup program.

When prompted for the host name, recall that on a command switch, the host name is limited to 28 characters; on a member switch to 31 characters. Do not use *-n*, where *n* is a number, as the last characters in a host name for any switch.

When prompted for the Telnet (virtual terminal) password, recall that it can be from 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

- Step 12** When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

- Step 13** When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

- Step 14** When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

- Step 15** After the initial configuration appears, verify that the addresses are correct.

- Step 16** If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

- Step 17** Start your browser, and enter the IP address of the new command switch.

- Step 18** From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.
-

Replacing a Failed Command Switch with Another Switch

To replace a failed command switch with a switch that is command-capable but not part of the cluster, follow these steps:

Step 1 Insert the new switch in place of the failed command switch, and duplicate its connections to the cluster members.

Step 2 Start a CLI session on the new command switch.

You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see the switch hardware installation guide.

Step 3 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
Switch#
```

Step 4 Enter the password of the *failed command switch*.

Step 5 Use the setup program to configure the switch IP information.

This program prompts you for IP address information and passwords. From privileged EXEC mode, enter **setup**, and press **Return**.

```
Switch# setup
--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
```

```
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
```

```
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
```

```
Would you like to enter basic management setup? [yes/no]:
```

Step 6 Enter **Y** at the first prompt.

The prompts in the setup program vary depending on the switch you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
```

or

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

Step 7 Respond to the questions in the setup program.

When prompted for the host name, recall that on a command switch, the host name is limited to 28 characters. Do not use *-n*, where *n* is a number, as the last characters in a host name for any switch.

When prompted for the Telnet (virtual terminal) password, recall that it can be from 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

Step 8 When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

Step 9 When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

- Step 10** When prompted, assign a name to the cluster, and press **Return**.
The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.
- Step 11** When the initial configuration displays, verify that the addresses are correct.
- Step 12** If the displayed information is correct, enter **Y**, and press **Return**.
If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.
- Step 13** Start your browser, and enter the IP address of the new command switch.
- Step 14** From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.
-

Recovering from Lost Member Connectivity

Some configurations can prevent the command switch from maintaining contact with member switches. If you are unable to maintain management contact with a member, and the member switch is forwarding packets normally, check for these conflicts:

- A member switch (Catalyst 3550, Catalyst 3500 XL, Catalyst 2955, Catalyst 2950, Catalyst 2940, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900) cannot connect to the command switch through a port that is defined as a network port.
- Catalyst 3500 XL, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900 member switches must connect to the command switch through a port that belongs to the same management VLAN.
- A member switch (Catalyst 3550, Catalyst 3500 XL, Catalyst 2950, Catalyst 2940, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900) connected to the command switch through a secured port can lose connectivity if the port is disabled because of a security violation.

Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the switch settings for speed (10 Mbps, 100 Mbps, and 1000 Mbps excluding GBIC ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize switch performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.



Note

If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

GBIC Module Security and Identification

Cisco-approved Course Wave Division Multiplexer (CWDM) Gigabit Interface Converter (GBIC) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When a CWDM GBIC module is inserted in the switch, the switch software reads the EEPROM to check the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the switch places the interface in an error-disabled state.

**Note**

If you are using a non-Cisco approved CWDM GBIC module, remove the GBIC module from the switch, and replace it with a Cisco-approved module.

After inserting a Cisco-approved GBIC module, use the **errdisable recovery cause gbic-invalid** global configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the switch brings the interface out of the error-disabled state and retries the operation. For more information about the **errdisable recovery** command, see the command reference for this release.

Diagnosing Connectivity Problems

This section describes how to troubleshoot connectivity problems:

- [Using Ping, page 36-11](#)
- [Using IP Traceroute, page 36-13](#)
- [Using Layer 2 Traceroute, page 36-14](#)

Using Ping

This section consists of this information:

- [Understanding Ping, page 36-11](#)
- [Executing Ping, page 36-12](#)

Understanding Ping

The switch supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply. Ping returns one of these 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.

Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets. For more information, see [Chapter 30, “Configuring IP Unicast Routing.”](#)

IP routing is disabled by default on all switches. If you need to enable or configure IP routing, see [Chapter 30, “Configuring IP Unicast Routing.”](#)

Beginning in privileged EXEC mode, use this command to ping another device on the network from the switch:

Command	Purpose
<code>ping [ip] {host address}</code>	Ping a remote host through IP or by supplying the host name or network address.



Note

Though other protocol keywords are available with the `ping` command, they are not supported in this release.

This example shows how to ping an IP host:

```
Switch# ping 172.20.52.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Switch#
```

[Table 36-1](#) describes the possible ping character output.

Table 36-1 Ping Output Display Characters

Character	Description
!	Each exclamation point means receipt of a reply.
.	Each period means the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

To terminate a ping session, enter the escape sequence (**Ctrl-^ X** by default). You enter the default by simultaneously pressing and releasing the **Ctrl**, **Shift**, and **6** keys, and then pressing the **X** key.

Using IP Traceroute

This section consists of this information:

- [Understanding IP Traceroute, page 36-13](#)
- [Executing IP Traceroute, page 36-13](#)

Understanding IP Traceroute

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.

Your switches can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the switch is the destination of the traceroute, it appears as the final destination in the traceroute output. Intermediate switches do not show up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate switch is a multilayer switch that is routing a particular packet, this switch shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC 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-to-live-exceeded message to the sender. Traceroute determines the address of the first hop by examining the source address field of the ICMP time-to-live-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-to-live-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 number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP port unreachable error to the source. Because all errors except port unreachable errors come from intermediate hops, the receipt of a port unreachable error means this message was sent by the destination.

Executing IP Traceroute

Beginning in privileged EXEC mode, follow this step to trace the path packets take through the network:

Command	Purpose
traceroute ip <i>host</i>	Trace the path packets take through the network by using IP.



Note

Though other protocol keywords are available with the **traceroute** privileged EXEC command, they are not supported in this release.

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 171.9.15.10

Type escape sequence to abort.
Tracing the route to 171.69.115.10

 0 172.2.52.1 0 msec 0 msec 4 msec
 1 172.2.1.203 12 msec 8 msec 0 msec
 2 171.9.16.6 4 msec 0 msec 0 msec
 3 171.9.4.5 0 msec 4 msec 0 msec
 4 171.9.121.34 0 msec 4 msec 4 msec
 5 171.9.15.9 120 msec 132 msec 128 msec
 6 171.9.15.10 132 msec 128 msec 128 msec
Switch#
```

The display shows the hop count, IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

Table 36-2 Traceroute Output Display Characters

Character	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output means that an access list is blocking traffic.
H	Host unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
U	Port unreachable.

To terminate a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). You enter the default by simultaneously pressing and releasing the **Ctrl**, **Shift**, and **6** keys, and then pressing the **X** key.

Using Layer 2 Traceroute

This section describes this information:

- [Understanding Layer 2 Traceroute, page 36-14](#)
- [Usage Guidelines, page 36-15](#)
- [Displaying the Physical Path, page 36-16](#)

Understanding 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.

The switch can only identify 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.

Usage Guidelines

These are the Layer 2 traceroute usage guidelines:

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to function 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 21, “Configuring CDP.”](#)

- A switch is reachable from another switch when you can test connectivity by using the **ping** privileged EXEC command. All switches in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command 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 addresses with the corresponding MAC addresses and the VLAN IDs.
 - 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.

Displaying the Physical Path

You can display physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

- **traceroute mac** [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]
- **traceroute mac ip** {*source-ip-address* | *source-hostname*} {*destination-ip-address* | *destination-hostname*} [**detail**]

For more information, see the command reference for this release.

Troubleshooting Power over Ethernet Switch Ports

This section consists of this information:

- [Disabled Port Caused by Power Loss, page 36-16](#)
- [Disabled Port Caused by False Link-Up, page 36-16](#)

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a Power over Ethernet (PoE) switch port and is powered by an AC power source loses power from the AC power source, the device might enter an error-disabled state. To recover from an error-disabled state, enter the **shutdown** interface configuration command followed by the **no shutdown** interface command. You can also configure automatic recovery on the switch to recover from the error-disabled state. The **errdisable recovery cause loopback** and the **errdisable recovery interval seconds** global configuration commands automatically take the interface out of the error-disabled state after the specified period of time.

Use these commands, described in the command reference for this release, to monitor the PoE port status:

- **show controllers power inline** privileged EXEC command
- **show power inline** privileged EXEC command
- **debug ilpower** privileged EXEC command

Disabled Port Caused by False Link-Up

If a Cisco powered device is connected to a port and you configure the port by using the **power inline never** interface configuration command, a false link-up can occur on the port, placing it into an error-disabled state. To take the port out of the error-disabled state, enter the **shutdown** and the **no shutdown** interface configuration commands.

You should not connect a Cisco powered device to a port that has been configured with the **power inline never** command.

Using Debug Commands

This section explains how you use the **debug** commands to diagnose and resolve internetworking problems. It contains this information:

- [Enabling Debugging on a Specific Feature, page 36-17](#)
- [Enabling All-System Diagnostics, page 36-18](#)
- [Redirecting Debug and Error Message Output, page 36-18](#)
- [Using the debug auto qos Command, page 36-18](#)



Caution

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.



Note

For complete syntax and usage information for specific **debug** commands, see the command reference for this release.

Enabling Debugging on a Specific Feature

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments. For example, beginning in privileged EXEC mode, enter this command to enable the debugging for EtherChannel:

```
Switch# debug etherchannel
```

The switch continues to generate output until you enter the **no** form of the command.

If you enable a **debug** command and no output appears, consider these possibilities:

- The switch might not be properly configured to generate the type of traffic that you want to monitor. Use the **show running-config** command to verify the configuration.
- Even if the switch is properly configured, it might not generate the type of traffic that you want to monitor during the particular period that debugging is enabled. Depending on the feature you are debugging, you can use commands such as the TCP/IP **ping** command to generate network traffic.

To disable debugging of EtherChannel, enter this command in privileged EXEC mode:

```
Switch# no debug etherchannel
```

Alternately, in privileged EXEC mode, you can enter the **undebug** form of the command:

```
Switch# undebug etherchannel
```

To display the state of each debugging option, enter this command in privileged EXEC mode:

```
Switch# show debugging
```

Enabling All-System Diagnostics

Beginning in privileged EXEC mode, enter this command to enable all-system diagnostics:

```
Switch# debug all
```



Caution

Because debugging output takes priority over other network traffic, and because the **debug all** privileged EXEC command generates more output than any other **debug** command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.

Redirecting Debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the console port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.



Note

Be aware that the debugging destination you use affects system overhead. Logging messages to the console produces very high overhead, whereas logging messages to a virtual terminal produces less overhead. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

For more information about system message logging, see [Chapter 25, “Configuring System Message Logging.”](#)

Using the debug auto qos Command

You can use the **debug auto qos** privileged EXEC command to display quality of service (QoS) commands that are automatically generated when automatic-QoS (auto-QoS) is enabled.

Beginning in privileged EXEC mode, follow these steps to display the QoS commands and enable auto-QoS for voice over IP (VoIP) within a QoS domain:

	Command	Purpose
Step 1	debug auto qos	Enable debugging for auto-QoS. When debugging is enabled, the switch displays the QoS commands that are automatically generated when auto-QoS is enabled or disabled.
Step 2	configure terminal	Enter global configuration mode.

	Command	Purpose
Step 3	<code>interface interface-id</code>	Specify the interface that is connected to a Cisco IP Phone, and enter interface configuration mode. You also can specify the uplink interface that is connected to another switch or router in the interior of the network.
Step 4	<code>auto qos voip {cisco-phone trust}</code>	Enable auto-QoS. The keywords have these meanings: <ul style="list-style-type: none"> • cisco-phone—If the interface is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the IP phone is detected. • trust—The uplink interface is connected to a trusted switch or router, and the VoIP classification in the ingress packet is trusted.
Step 5	<code>end</code>	Return to privileged EXEC mode.
Step 6	<code>show auto qos interface interface-id</code>	Verify your entries. This command displays the auto-QoS configuration that was initially applied; it does not display any user changes to the configuration that might be in effect.

For more information about auto-QoS, see the [“Configuring Auto-QoS” section on page 28-17](#).

This example shows how to display the QoS commands that are automatically generated when auto-QoS is enabled:

```
Switch# debug auto qos
AutoQoS debugging is on
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# auto qos voip cisco-phone
```

Using the show forward Command

The output from the **show forward** privileged EXEC command has some useful information about the disposition of a packet entering an interface. Depending upon the parameters entered about the packet, the output shows lookup table results, maps and masks used to calculate forwarding destinations, bitmaps, and egress information.



Note

For more syntax and usage information for the **show forward** command, see the command reference for this release.

This is an example of the output from the **show forward** privileged EXEC command for Fast Ethernet port 8, where VLAN ID, source and destination MAC addresses, and source and destination IP addresses were specified.

```
Switch# show forward fastethernet0/8 vlan 8 0000.1111.2222 0022.3355.9800 ip 8.8.8.10
4.4.4.33 255
signature:00000007, comparison ind:10, control info:2000941A control map:00000000
vlan:8, vlanid entry:000C0012 00000000 00000000 04620000
vlan:8, vlanid entry:000C0012 00000000 00000000 04620000

lookup key                               bk adata   rawoff seccff sec
qos      940808080A04040421 800000000000FF0000 0 00000000 006304 004064 4
acl      940808080A04040421 800000000000FF0000 1 00000082 045408 002016 1
learn    187008000011112222 801008002233559800 0 80010003 002176 002176 0
forw     187008000011112222 801008002233559800 1 40020000 043328 010560 5

bridgeDestMap: 00000000 00000000 0000FFFF FFFFFFFC7
vlanMask:      00000000 00000000 0000FFFF FFFFFFFE7F
portMask:      00000000 00000000 00000000 00000080
sourceMask:    00000000 00000000 00000000 00000000
globalMap:     00000000 00000000 00000000 00000000
globalMask:    00000000 00000000 0002FFFF EFFFFFFC03
forwMap:       00000000 00000000 00000000 00000100

frame notifies:
src u_dat vlan fl q-map
2 00 8 00 00000000 00000000 00000000 00000100
Egress q 8
signature:00000007, comparison ind:10, control info:2000941A control map:00000000
vlan:8, vlanid entry:000C0012 00000000 00000000 04620000
FastEthernet0/9 vlan 8, dst 0022.3355.9800 src 0000.1111.2222, cos 0x0, dscp 0x0
```

Much of this information is useful mainly for Technical Support personnel, who have access to detailed information about the switch application-specific integrated circuits (ASICs). However, you can look at the *Egress q* section to get information about the output interface. There is an egress section for each separate destination port. The important information is in the line containing the name of the output interface, output VLAN ID, and rewritten destination MAC address for the frame. The example shows that the output interface is Fast Ethernet port 9 and the output VLAN is VLAN 8 and shows the rewritten source and destination MAC address for the frame.

If the output interface is a trunk port that needs to send multiple copies of the frame on different VLANs (for example, for IP multicast frames), several lines might contain the same output interface name, but different output VLANs. If output security access control lists (ACLs) are present, it is possible that one or more of these *Egress q* sections will not contain a line listing an output port. This happens when the output ACL denies the packet.

When the CPU is one of the destinations for a packet, a *Cpu q* section appears, followed by a queue name. This name should correspond to one of the queue names in the output from the **show controllers cpu-interface** privileged EXEC command, where statistics appear for the number of packets received at each queue.

This is an example of the *Cpu q* section display:

```
Cpu q:100 - routing queue
```

Using the crashinfo File

The crashinfo file saves information that helps Cisco technical support representatives to debug problems that caused the software image to fail (crash). The switch writes the crash information to the console at the time of the failure, and the file is created the next time you boot the image after the failure (instead of while the system is failing).

The information in the file includes the software image name and version that failed, a dump of the processor registers, and a stack trace. You can give this information to the Cisco technical support representative by using the **show tech-support** privileged EXEC command.

All crashinfo files are kept in this directory on the flash file system:

flash:/crashinfo/crashinfo_*n* where *n* is a sequence number.

Each new crashinfo file that is created uses a sequence number that is larger than any previously existing sequence number, so the file with the largest sequence number describes the most recent failure. Version numbers are used instead of a timestamp because the switches do not include a real-time clock. You cannot change the name of the file that the system will use when it creates the file. However, after the file is created, you can use the **rename** privileged EXEC command to rename it, but the contents of the renamed file will not be displayed by the **show stacks** or the **show tech-support** privileged EXEC command. You can delete crashinfo files by using the **delete** privileged EXEC command.

You can display the most recent crashinfo file (that is, the file with the highest sequence number at the end of its filename) by entering the **show stacks** or the **show tech-support** privileged EXEC command. You also can access the file by using any command that can copy or display files, such as the **more** or the **copy** privileged EXEC command.

