traceroute mac

To display the Layer 2 path taken by the packets from the specified source to the specified destination, use the **traceroute mac** command.

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

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

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

Syntax Description

source-mac-address	MAC address of the source switch in hexadecimal format.
destination-mac-address	MAC address of the destination switch in hexadecimal format.
interface *type*	Specifies the interface where the MAC address resides; valid values are **FastEthernet**, **GigabitEthernet**, and **Port-channel**.
interface-number	Module and port number or the port-channel number; valid values for the port channel are from 1 to 282.
vlan *vlan-id*	(Optional) Specifies the VLAN on which to trace the Layer 2 path that the packets take from the source switch to the destination switch; valid values are from 1 to 4094.
detail	(Optional) Displays detailed information about the Layer 2 trace.
ip	Specifies the IP address where the MAC address resides.
source-ip-address	IP address of the source switch as a 32-bit quantity in dotted-decimal format.
source-hostname	IP hostname of the source switch.
destination-ip-address	IP address of the destination switch as a 32-bit quantity in dotted-decimal format.
destination-hostname	IP hostname of the destination switch.

Command Default

This command has no default settings.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Do not use leading zeros when entering a VLAN ID.

You must enable CDP on all of the switches in the network. Do not disable CDP so that Layer 2 traceroute can function properly.

When the switch detects a device in the Layer 2 path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

The maximum number of hops identified in the path is ten.

Layer 2 traceroute supports unicast traffic only. If you specify a multicast source or destination MAC address, the physical path is not identified, and a message appears.

The **traceroute mac** command output shows the Layer 2 path when the specified source and destination addresses belong to the same VLAN. If you specify source and destination addresses that belong to different VLANs, the Layer 2 path is not identified, and a 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 a 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 utility terminates at that hop and displays an error message.

This feature is not supported in Token Ring VLANs.

Examples

This example shows how to display detailed information about the Layer 2 path:

```
Router# traceroute mac 0001.0000.0204 0001.0000.0304 detail
Source 0001.0000.0204 found on VAYU[WS-C6509] (2.1.1.10)
1 VAYU / WS-C6509 / 2.1.1.10 :
Gi6/1 [full, 1000M] => Po100 [auto, auto]
2 PANI / WS-C6509 / 2.1.1.12 :
Po100 [auto, auto] => Po110 [auto, auto]
3 BUMI / WS-C6509 / 2.1.1.13 :
Po110 [auto, auto] => Po120 [auto, auto]
4 AGNI / WS-C6509 / 2.1.1.11 :
Po120 [auto, auto] => Gi8/12 [full, 1000M]
Destination 0001.0000.0304 found on AGNI[WS-C6509] (2.1.1.11)
Layer 2 trace completed.
Router#
```

This example shows the output when the switch is not connected to the source switch:

```
Router# traceroute mac 0000.0201.0501 0000.0201.0201 detail
Source not directly connected, tracing source .....
Source 0000.0201.0501 found on con5[WS-C6509] (2.2.5.5)
con5 / WS-C6509 / 2.2.5.5 :
        Fa0/1 [auto, auto] =>Gi0/1 [auto, auto]
con1 / WS-C6509 / 2.2.1.1 :
        Gi0/1 [auto, auto] =>Gi0/2 [auto, auto]
con2 / WS-C6509 / 2.2.2.2 :
        Gi0/2 [auto, auto] =>Fa0/1 [auto, auto]
Destination 0000.0201.0201 found on con2[WS-C6509] (2.2.2.2)
Layer 2 trace completed.
Router#
```

This example shows the output when the switch cannot find the destination port for the source MAC address:

```
Router# traceroute mac 0000.0011.1111 0000.0201.0201
Error:Source Mac address not found.
Layer2 trace aborted.
Router#
```

This example shows the output when the source and destination devices are in different VLANs:

```
Router# traceroute mac 0000.0201.0601 0000.0301.0201
```

```
Error:Source and destination macs are on different vlans.
Layer2 trace aborted.
Router#
```

This example shows the output when the destination MAC address is a multicast address:

```
Router# traceroute mac 0000.0201.0601 0100.0201.0201
Invalid destination mac address
Router#
```

This example shows the output when the source and destination switches belong to multiple VLANs:

```
Router# traceroute mac 0000.0201.0601 0000.0201.0201
Error:Mac found on multiple vlans.
Layer2 trace aborted.
Router#
```

This example shows how to display the Layer 2 path by specifying the interfaces on the source and destination switches:

```
Router# traceroute mac interface fastethernet0/1 0000.0201.0601 interface fastethernet0/3 0000.0201.0201
Source 0000.0201.0601 found on con6[WS-C6509] (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-C6509] (2.2.2.2)
Layer 2 trace completed
Router#
```

This example shows how to display detailed traceroute information:

```
Router# 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-C6509] (2.2.6.6)
con6 / WS-C6509 / 2.2.6.6 :
        Fa0/1 [auto, auto] =>Fa0/3 [auto, auto]
con5 / WS-C6509 / 2.2.5.5 :
        Fa0/3 [auto, auto] =>Gi0/1 [auto, auto]
con1 / WS-C6509 / 2.2.1.1 :
        Gi0/1 [auto, auto] =>Gi0/2 [auto, auto]
con2 / WS-C6509 / 2.2.2.2 :
        Gi0/2 [auto, auto] =>Fa0/1 [auto, auto]
Destination 0000.0201.0201 found on con2[WS-C6509] (2.2.2.2)
Layer 2 trace completed.
Router#
```

This example shows how to display the Layer 2 path by specifying the source and destination hostnames:

```
Router# traceroute mac ip con6 con2
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
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
Layer 2 trace completed
Router#
```

This example shows the output when ARP cannot associate the source IP address with the corresponding MAC address:

```
Router# traceroute mac ip 2.2.66.66 2.2.77.77
Arp failed for destination 2.2.77.77.
Layer2 trace aborted.
Router#
```

track interface

To configure an interface to be tracked and to enter tracking configuration mode, use the **track interface** command in global configuration mode. To remove the tracking, use the **no** form of this command.

track *object-number* **interface** *type number* {**line-protocol** | **ip routing**}

no track *object-number* **interface** *type number* {**line-protocol** | **ip routing**}

Syntax Description

object-number	Object number that represents the interface to be tracked; valid values are from 1 to 500.
type number	Interface type and number to be tracked.
line-protocol	Tracks the state of the interface line protocol.
ip routing	Tracks if IP routing is enabled, if an IP address is configured on the interface, and if the interface state is up before reporting to the tracking client that the interface is up.

Command Default

No interface is tracked.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command reports a state value to clients. A tracked IP-routing object is considered up when the following exists:

- IP routing is enabled and active on the interface.
- The interface line-protocol state is up.
- The interface IP address is known. The IP address is configured or received through the Dynamic Host Configuration Protocol (DHCP) or IP Control Protocol (IPCP) negotiation.

Interface IP routing will go down when one of the following exists:

- IP routing is disabled globally.
- The interface line-protocol state is down.
- The interface IP address is unknown. The IP address is not configured or received through DHCP or IPCP negotiation.

No space is required between the *type number* values.

Tracking the IP-routing state of an interface using the **track interface ip routing** command can be more useful in some situations than just tracking the line-protocol state using the **track interface line-protocol** command, especially on interfaces for which IP addresses are negotiated. For example, on a serial interface that uses the Point-to-Point Protocol (PPP), the line protocol could be up [link control protocol (LCP) negotiated successfully], but IP could be down (IPCP negotiation failed).

The **track interface ip routing** command supports the tracking of an interface with an IP address acquired through any of the following methods:

- Conventional IP address configuration
- PPP/IPCP
- DHCP
- Unnumbered interface

Examples

This example shows how to configure the tracking process to track the IP-routing capability of serial interface 1/0:

```
Router(config)# track 1 interface serial1/0 ip routing
Router(config)#
```

Related Commands

Command	Description
show track	Displays HSRP tracking information.

transceiver type all monitoring

To enable monitoring on all transceivers, use the **transceiver type all monitoring** command. To disable monitoring, use the **no** form of this command.

transceiver type all monitoring

no transceiver type all monitoring

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

You can use the **transceiver type all monitoring** command to enable monitoring (for example, collecting DOM information and evaluating threshold violations) for all transceiver types.

Note The **no transceiver type all monitoring** command overrides the **snmp-server enable traps tranceiver type all** command and will not permit the generation of SNMP traps.

Examples

This example shows how to enable monitoring for all transceiver types:

```
Router(config)# transceiver type all monitoring
Router(config)#
```

This example shows how to disable monitoring for all transceiver types:

```
Router(config)# no transceiver type all monitoring
Router(config)#
```

Related Commands

Command	Description
snmp-server enable traps transceiver type all	Enables all supported SNMP transceiver traps for all transceiver types.

tunnel udlr address-resolution

To enable the forwarding of the ARP and NHRP over a UDL, use the **tunnel udlr address-resolution** command. To disable forwarding, use the **no** form of this command.

tunnel udlr address-resolution

no tunnel udlr address-resolution

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command is supported on the following modules only:

- WS-X6704-10GE 4-port 10-Gigabit Ethernet
- WS-X6816-GBIC 16-port Gigabit Ethernet
- WS-X6516A-GBIC 16-port Gigabit Ethernet
- WS-X6516-GBIC 16-port Gigabit Ethernet

UDLR back-channel tunnels do not support IPv6.

This command is supported on the send-only tunnel interface of a downstream router only.

You cannot configure software-based UDE on non-physical interfaces.

An ARP address resolution request that is received from the upstream router on the UDL (Ethernet interface 0) is replied to over the send-only tunnel of the receiver. An ARP request may be sent by the downstream router over the send-only tunnel, and the response is received over the UDL.

Examples

This example shows how to enable ARP and NHRP forwarding over a send-only tunnel:

```
Router(config-if)# tunnel udlr address-resolution
Router(config-if)#
```

Related Commands

Command	Description
show ip igmp udlr	Displays UDLR information for the connected multicast groups on the interfaces that have a UDL helper address configured.
tunnel udlr receive-only	Configures a unidirectional GRE tunnel to act as a back channel that can receive messages from an interface that is configured for unidirectional link routing.

tunnel udlr receive-only

To configure a unidirectional GRE tunnel to act as a back channel that can receive messages from an interface that is configured for unidirectional link routing, use the **tunnel udlr receive-only** command. To remove the tunnel, use the **no** form of this command.

tunnel udlr receive-only *interface-type interface-number*

no tunnel udlr receive-only *interface-type interface-number*

Syntax Description

interface-type *interface-number*	Interface type and number.

Command Default

No UDLR tunnel is configured.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command is supported on the following modules only:

- WS-X6704-10GE 4-port 10-Gigabit Ethernet
- WS-X6816-GBIC 16-port Gigabit Ethernet
- WS-X6516A-GBIC 16-port Gigabit Ethernet
- WS-X6516-GBIC 16-port Gigabit Ethernet

The UDLR back-channel tunnels do not support IPv6.

Use this command to configure a router that has a unidirectional interface with send-only capabilities. For example, you can use this command if you have traffic traveling through a satellite.

The *interface-type* and *interface-number* arguments must match the send-only interface type and number specified by the **interface** command.

The *interface-type* and *interface-number* arguments must match the unidirectional send-only interface type and number specified by the **interface** command. When the packets are received over the tunnel, the upper layer protocols treat the packets as if they are received over the unidirectional send-only interface.

You must configure the **tunnel udlr send-only** command at the opposite end of the tunnel.

For a description of the **ip igmp unidirectional-link** command, refer to the *Cisco IOS Release 12.2 Command Reference*.

Examples

This example shows how to configure a unidirectional GRE tunnel to act as a back channel that can receive messages from an interface that is configured for unidirectional link routing:

```
Router(config-if)# tunnel udlr receive-only serial 0
Router(config-if)#
```

Related Commands

Command	Description
interface	Selects an interface to configure and enters interface configuration mode.
ip igmp unidirectional-link	Configures an interface to be unidirectional and enables it for IGMP UDLR.
show ip igmp udlr	Displays UDLR information for the connected multicast groups on the interfaces that have a UDL helper address configured.
tunnel udlr send-only	Configures a unidirectional GRE tunnel to act as a back channel that can send messages from an interface that is configured for unidirectional link routing.

tunnel udlr send-only

To configure a unidirectional GRE tunnel to act as a back channel that can send messages from an interface that is configured for unidirectional link routing, use the **tunnel udlr send-only** command. To remove the tunnel, use the **no** form of this command.

tunnel udlr send-only *interface-type interface-number*

no tunnel udlr send-only *interface-type interface-number*

Syntax Description

interface-type interface-number	Interface type and number.

Command Default

No UDLR tunnel is configured.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command is supported on the following modules only:

- WS-X6704-10GE 4-port 10-Gigabit Ethernet
- WS-X6816-GBIC 16-port Gigabit Ethernet
- WS-X6516A-GBIC 16-port Gigabit Ethernet
- WS-X6516-GBIC 16-port Gigabit Ethernet

The UDLR back-channel tunnels do not support IPv6.

Use this command to configure a router that has a unidirectional interface with receive-only capabilities. The UDLR tunnel will act as a back channel. For example, you can use this command if you have traffic traveling through a satellite.

The *interface-type* and *interface-number* arguments must match the unidirectional receive-only interface type and number specified by the **interface** command.When packets are sent by the upper layer protocols over the interface, they are redirected and sent over this GRE tunnel.

The *interface-type* and *interface-number* arguments must match the receive-only interface type and number specified by the **interface** command.

You must configure the **tunnel udlr receive-only** command at the opposite end of the tunnel.

Examples

This example shows how to configure a unidirectional GRE tunnel to act as a back channel that can send messages from an interface that is configured for unidirectional link routing:

```
Router(config-if)# tunnel udlr send-only serial 1
Router(config-if)#
```

Related Commands

Command	Description
interface	Selects an interface to configure and enters interface configuration mode.
show ip igmp udlr	Displays UDLR information for the connected multicast groups on the interfaces that have a UDL helper address configured.
tunnel udlr address-resolution	Enables the forwarding of the ARP and NHRP over a UDL.
tunnel udlr receive-only	Configures a unidirectional GRE tunnel to act as a back channel that can receive messages from an interface that is configured for unidirectional link routing.

udld

To enable aggressive or normal mode in UDLD and set the configurable message time, use the **udld** command. To disable aggressive or normal mode in UDLD, use the **no** form of this command.

udld {**enable** | **aggressive**}

no udld {**enable** | **aggressive**}

udld message time *message-timer-time*

no udld message time

Syntax Description

udld enable	Enables UDLD in normal mode by default on all fiber interfaces.
udld aggressive	Enables UDLD in aggressive mode by default on all fiber interfaces.
message time *message-timer-time*	Sets the period of time between UDLD probe messages on ports that are in advertisement mode and are currently determined to be bidirectional; valid values are from 7 to 90 seconds.

Command Default

The defaults are as follows:

- UDLD is disabled on all fiber interfaces.
- *message-timer-time* is 15 seconds.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Use the **no** form of this command to do the following:

- Disable normal-mode UDLD on all fiber ports by default.
- Disable aggressive-mode UDLD on all fiber ports by default.
- Disable the message timer.

If you enable aggressive mode, after all the neighbors of a port age out either in the advertisement or in the detection phase, UDLD restarts the linkup sequence to resynchronize with any potentially out-of-sync neighbor and shuts down the port if the message from the link is still undetermined.

This command affects fiber interfaces only. Use the **udld port** command in interface-configuration mode to enable UDLD on other interface types.

Examples

This example shows how to enable UDLD on all fiber interfaces:

```
Router (config)# udld enable
Router (config)#
```

Related Commands

Command	Description
show udld	Displays the administrative and operational UDLD status.
udld port	Enables UDLD on the interface or enables UDLD in aggressive mode on the interface.

udld port

To enable UDLD on the interface or enable UDLD in aggressive mode on the interface, use the **udld port** command. To return to the default settings, use the **no** form of this command.

udld port [**aggressive**]

no udld port [**aggressive**]

Syntax Description

aggressive	(Optional) Enables UDLD in aggressive mode on this interface; see the "Usage Guidelines" section for additional information.

Command Default

The defaults are as follows:

- Fiber interfaces are in the state of the global **udld** (**enable** or **aggressive**) command.
- Nonfiber interfaces have UDLD disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command does not appear in the CLI unless a GBIC is in the port that you are trying to enable.

Use the **udld port** and **udld port aggressive** commands on fiber ports to override the setting of the global **udld** (**enable** or **aggressive**) command. Use the **no** form on fiber ports to remove this setting and return control of UDLD enabling back to the global **udld** command, or in the case of nonfiber ports, to disable UDLD.

If you enable aggressive mode, after all the neighbors of a port age out either in the advertisement or in the detection phase, UDLD restarts the linkup sequence to resynchronize with any potentially out-of-sync neighbor and shuts down the port if the message from the link is still undetermined.

If the port changes from fiber to nonfiber or nonfiber to fiber, all configurations are maintained because the platform software detects a change of module or a GBIC change.

Examples

This example shows how to cause any port interface to enable UDLD regardless of the current global **udld** setting:

```
Router (config-if)# udld port
Router (config-if)#
```

This example shows how to cause any port interface to enable UDLD in aggressive mode regardless of the current global **udld** (**enable** or **aggressive**) setting:

```
Router (config-if)# udld port aggressive
Router (config-if)#
```

This example shows how to cause a fiber port interface to disable UDLD regardless of the current global **udld** setting:

```
Router (config-if)# no udld port
Router (config-if)#
```

Related Commands

Command	Description
show udld	Displays the administrative and operational UDLD status.
udld	Enables aggressive or normal mode in UDLD and sets the configurable message time.

udld reset

To reset all the ports that are shut down by UDLD and permit traffic to begin passing through them again (although other features, such as spanning tree, PAgP, and DTP, will behave normally if enabled), use the **udld reset** command.

udld reset

Syntax Description

This command has no keywords or arguments.

Command Default

This command has no default settings.

Command Modes

EXEC mode

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

If the interface configuration is still enabled for UDLD, these ports will begin to run UDLD again and may shut down for the same reason if the reason for the shutdown has not been corrected.

Examples

This example shows how to reset all ports that are shut down by UDLD:

```
Router# udld reset
Router#
```

Related Commands

Command	Description
show udld	Displays the administrative and operational UDLD status.

udp-port

To change the UDP port numbers to which a test sender sends test packets or a test receiver sends status reports, use the **udp-port** command. To remove the port numbers, use the **no** form of this command.

udp-port [**test-packet** *port-number*] [**status-report** *port-number*]

no udp-port [**test-packet** *port-number*] [**status-report** *port-number*]

Syntax Description

test-packet *port-number*	(Optional) Specifies the UDP port number to which test packets are sent by a test sender.
status-report *port-number*	(Optional) Specifies the UDP port number to which status reports are sent by a test receiver.

Command Default

The defaults are as follows:

- **test-packet** *port-number*—16384, the minimum value of an audio port
- **status-report** *port-number*—65535, the maximum value of a video port

Command Modes

Manager configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command is supported on the following modules only:

- WS-X6704-10GE 4-port 10-Gigabit Ethernet
- WS-X6816-GBIC 16-port Gigabit Ethernet
- WS-X6516A-GBIC 16-port Gigabit Ethernet
- WS-X6516-GBIC 16-port Gigabit Ethernet

The **test-packet** *port-number* must be even if the packets are RTP encapsulated.

The **status-report** *port-number* must be odd if the packets are RTP encapsulated.

Examples

This example shows how to change the UDP port number to which test packets are targeted to 20000:

```
Router(config-mrm-manager)# udp-port test-packet 20000
Router(config-mrm-manager)#
```

Related Commands

Command	Description
ip mrm	Configures an interface to operate as a test sender or test receiver, or both, for MRM.

undelete

To recover a file that is marked “deleted” on a flash file system, use the **undelete** command.

undelete *index* [*filesystem***:**]

Syntax Description

index	Number to index the file in the **dir** command output; valid values are from 1 to 1024.
*filesystem***:**	(Optional) File system containing the file to undelete, followed by a colon; valid values are **bootflash:**, **disk0:**, **disk1:** , **flash:**, **slot0:**, or **sup-bootflash:**.

Command Default

The default file system is specified when you enter the **cd** command.

Command Modes

EXEC mode

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

On Class A flash file systems, when you delete a file, Cisco IOS software marks the file as deleted but does not erase the file. This command allows you to recover a deleted file on a specified flash-memory device. You must undelete a file by its index because you could have multiple deleted files with the same name. For example, the “deleted” list could contain multiple configuration files with the name router-config. You undelete by the index to indicate which of the many router-config files from the list to undelete. Use the **dir** command to learn the index number of the file that you want to undelete.

bootflash:, **flash:**, **disk0:**, **disk1:**, and **sup-bootflash:** are Class A file systems.

You cannot undelete a file if a valid (undeleted) file with the same name exists. Instead, you first delete the existing file and then undelete the file that you want. For example, if you had an undeleted version of the router-config file and you wanted to use a previous, deleted version instead, you could not simply undelete the previous version by index. You must first delete the existing router-config file and then undelete the previous router-config file by index. You can delete and undelete a file up to 15 times.

On Class A flash file systems, if you try to recover the configuration file that is pointed to by the CONFIG_FILE environment variable, you are prompted to confirm recovery of the file. This prompt reminds you that the CONFIG_FILE environment variable points to an undeleted file. To delete all files that are marked “deleted” on a flash-memory device permanently, use the **squeeze** command in EXEC mode.

Examples

This example shows how to recover the deleted file whose index number is 1 to the flash PC card that is inserted in disk 0:

```
Router# undelete 1 disk0:
Router#
```

Related Commands

Command	Description
delete	Deletes a file from a flash memory device or NVRAM.
dir	Displays a list of files on a file system.
squeeze	Deletes flash files permanently by squeezing a flash file system.

unidirectional

To configure the software-based UDE, use the **unidirectional** command. To remove the software-based UDE configuration, use the **no** form of this command.

unidirectional {**send-only** | **receive-only**}

no unidirectional

Syntax Description

send-only	Specifies that the unidirectional transceiver transmits traffic only.
receive-only	Specifies that the unidirectional transceiver receives traffic only.

Command Default

UDE is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

UDE is supported on the interfaces of these switching modules:

- WS-X6704-10GE 4-port 10-Gigabit Ethernet
- WS-X6816-GBIC 16-port Gigabit Ethernet
- WS-X6516A-GBIC 16-port Gigabit Ethernet
- WS-X6516-GBIC 16-port Gigabit Ethernet

You do not need to configure software-based UDE on ports where you implement hardware-based UDE.

If an interface is configured with Unidirectional Ethernet or has a receive-only transceiver, UDLD is operationally disabled. Use the **show udld** command to display the configured and operational states of this interface.

When you apply the UDE configuration to an interface, the following warning message is displayed:

```
Warning!
Enable port unidirectional mode will automatically disable port udld. You must manually
ensure that the unidirectional link does not create a spanning tree loop in the network.

Enable l3 port unidirectional mode will automatically disable ip routing on the port. You
must manually configure static ip route and arp entry in order to route ip traffic.
```

Examples

This example shows how to configure 10-Gigabit Ethernet port 1/1 as a UDE send-only port:

```
Router(config-if)# unidirectional send-only

Warning!
```

```
Enable port unidirectional mode will automatically disable port udld. You must manually
ensure that the unidirectional link does not create a spanning tree loop in the network.


Enable l3 port unidirectional mode will automatically disable ip routing on the port. You
must manually configure static ip route and arp entry in order to route ip traffic.
Router(config-if)#
```

This example shows how to configure 10-Gigabit Ethernet port 1/2 as a UDE receive-only port:

```
Router(config-if)# unidirectional receive-only

Warning!

Enable port unidirectional mode will automatically disable port udld. You must manually
ensure that the unidirectional link does not create a spanning tree loop in the network.

Enable l3 port unidirectional mode will automatically disable ip routing on the port. You
must manually configure static ip route and arp entry in order to route ip traffic.
Router(config-if)#
```

Related Commands

Command	Description
show interfaces status	Displays the interface status or a list of interfaces in an error-disabled state on LAN ports only.
show interfaces unidirectional	Displays the operational state of an interface with a receive-only transceiver.

upgrade rom-monitor

To set the execution preference on a ROMMON, use the **upgrade rom-monitor** command.

upgrade rom-monitor {**slot** *num*} {**sp** | **rp**} {**file** *filename*}

upgrade rom-monitor {**slot** *num*} {**sp** | **rp**} {{**invalidate** | **preference**} {**region1** | **region2**}}

Syntax Description

slot *num*	Specifies the slot number of the ROMMON to be upgraded.
sp	Upgrades the ROMMON of the switch processor.
rp	Upgrades the ROMMON of the route processor.
file *filename*	Specifies the name of the SREC file; see the "Usage Guidelines" section for valid values.
invalidate	Invalidates the ROMMON of the selected region.
preference	Sets the execution preference on a ROMMON of the selected region.
region1	Selects the ROMMON in region 1.
region2	Selects the ROMMON in region 2.

Command Default

This command has no default settings.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Caution If you enter the **upgrade rom-monitor** command with no parameters, service may be interrupted.

Caution If you enter the **upgrade rom-monitor** command from a Telnet session instead of a console connection, service may be interrupted.

The **slot** *num* is required for this command to function properly.

The **sp** or **rp** keyword is required if you installed a supervisor engine in the specified slot.

Valid values for **file** *filename* include the following:

- **bootflash:**
- **disk0:**
- **disk1:**

Examples

This example shows how to upgrade the new ROMMON image to the flash device:

```
Router# upgrade rom-monitor slot 1 sp file tftp://dirt/tftpboot-users/A2_71059.srec
ROMMON image upgrade in progress
   Erasing flash
   Programming flash
   Verifying new image
   ROMMON image upgrade complete
   The card must be reset for this to take effect
Router#
```

Related Commands

Command	Description
show rom-monitor	Displays the ROMMON status.

username secret

To establish a username-based authentication system, use the **username secret** command.

username *name* **secret** {**0** | **5**} *password*

Syntax Description

name	User ID.
secret 0 | **5**	Specifies the secret; valid values are **0** (text immediately following is not encrypted) and **5** (text immediately following is encrypted using an MD5-type encryption method).
password	Password.

Command Default

No username-based authentication system is established.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Use this command to enable Enhanced Password Security for the specified, unretrievable username. This command enables MD5 encryption on the password. MD5 encryption is a strong encryption method. You cannot use MD5 encryption with protocols, such as CHAP, that require clear-text passwords.

This command can be useful for defining usernames that get special treatment. For example, you can use this command to define an “info” username that does not require a password but connects the user to a general-purpose information service.

The **username secret** command provides a username and/or a secret authentication for login purposes only. The *name* argument can be one word only. White spaces and quotation marks are not allowed. You can use multiple **username secret** commands to specify options for a single user.

Examples

This example shows how to configure a username xena and enter an MD5 encrypted text string that is stored as the username password:

```
Router(config)# username xena secret 5 $1$feb0$a104Qd9UZ./Ak00KTggPD0
Router(config)#
```

Related Commands

Command	Description
enable password	Sets a local password to control access to various privilege levels.
enable secret	Specifies an additional layer of security over the **enable password** command.