

IP Commands

The Internet Protocol (IP) is a packet-based protocol used to exchange data over computer networks. IP handles addressing, fragmentation, reassembly, and protocol demultiplexing. It is the foundation on which all other Internet protocols, collectively referred to as the Internet Protocol suite, are built. IP is a network-layer protocol that contains addressing information and some control information that allows data packets to be routed.

The Transmission Control Protocol (TCP) is built upon the IP layer. TCP is a connection-oriented protocol that specifies the format of data and acknowledgments used in the transfer of data. TCP also specifies the procedures that the computers use to ensure that the data arrives correctly. TCP allows multiple applications on a system to communicate concurrently because it handles all demultiplexing of the incoming traffic among the application programs.

Use the commands in this chapter to configure and monitor IP networks. For IP protocol configuration information and examples, refer to the chapter “Configuring IP” in the *Access and Communication Servers Configuration Guide*.

access-class

To restrict incoming and outgoing connections between a particular virtual terminal line (into a Cisco device) and the addresses in an access list, use the **access-class** line configuration command. To remove access restrictions, use the **no** form of this command.

```
access-class access-list-number { in | out }  
no access-class access-list-number { in | out }
```

Syntax Description

<i>access-list-number</i>	Number of an access list. This is a decimal number from 1 through 99.
in	Restricts incoming connections between a particular Cisco device and the addresses in the access list.
out	Restricts outgoing connections between a particular Cisco device and the addresses in the access list.

Default

No access lists are defined.

Command Mode

Line configuration

Usage Guidelines

Remember to set *identical restrictions* on all the virtual terminal lines because a user can connect to any of them.

To display the access lists for a particular terminal line, use the **show line EXEC** command and specify the line number.

Examples

The following example defines an access list that permits only hosts on network 192.89.55.0 to connect to the virtual terminal ports on the communication server:

```
access-list 12 permit 192.89.55.0 0.0.0.255  
line 1 5  
access-class 12 in
```

The following example defines an access list that denies connections to networks other than network 36.0.0.0 on terminal lines 1 through 5:

```
access-list 10 permit 36.0.0.0 0.255.255.255  
line 1 5  
access-class 10 out
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show line †

access-list (standard)

To define a standard IP access list, use the standard version of the **access-list** global configuration command. To remove a standard access lists, use the **no** form of this command.

```
access-list access-list-number { deny | permit } source [source-wildcard]  
no access-list access-list-number
```



Caution Enhancements to this command are backward compatible; migrating from existing releases to Cisco IOS Release 10.3 will convert your access lists automatically. However, releases prior to Cisco IOS Release 10.3 are not upwardly compatible with these enhancements. Therefore, if you save an access list with these images and then use software prior to Cisco IOS Release 10.3, the resulting access list will not be interpreted correctly. *This could cause you severe security problems.* Save your old configuration file before booting these images.

Syntax Description

<i>access-list-number</i>	Number of an access list. This is a decimal number from 1 through 99.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
<i>source</i>	Number of the network or host from which the packet is being sent. There are two alternative ways to specify the source: <ul style="list-style-type: none"> • Use a 32-bit quantity in four-part dotted-decimal format. • Use the keyword any as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255.
<i>source-wildcard</i>	(Optional) Wildcard bits to be applied to the <i>source</i> . There are two alternative ways to specify the source wildcard: <ul style="list-style-type: none"> • Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore. • Use the keyword any as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255.

Default

The access list defaults to an implicit deny statement for everything. The access list is always terminated by an implicit deny statement for everything.

Command Mode

Global configuration

Usage Guidelines

Plan your access conditions carefully and be aware of the implicit deny statement at the end of the access list.

You can use access lists to control the transmission of packets on an interface, control virtual terminal line access, and restrict the contents of routing updates.

Use the **show access-lists** EXEC command to display the contents of all access lists.

Use the **show ip access-list** EXEC command to display the contents of one access list.

Examples

The following example of a standard access list allows access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements will be rejected.

```
access-list 1 permit 192.5.34.0 0.0.0.255
access-list 1 permit 128.88.0.0 0.0.255.255
access-list 1 permit 36.0.0.0 0.255.255.255
! (Note: all other access implicitly denied)
```

To specify a large number of individual addresses more easily, you can omit the wildcard if it is all zeroes. Thus, the following two configuration commands are identical in effect:

```
access-list 2 permit 36.48.0.3
access-list 2 permit 36.48.0.3 0.0.0.0
```

To specify a large number of individual addresses more easily, you can omit the address mask; that is, all zeros from the **access-list** command. Thus, the following two configuration commands are identical in effect:

```
access-list 2 permit 36.48.0.3
access-list 2 permit 36.48.0.3 0.0.0.0
```

Related Commands

access-class

access-list (extended)

distribute-list

ip access-group

priority-list

queue-list

show access-lists

show ip access-list

access-list (extended)

To define an extended IP access list, use the extended version of the **access-list** global configuration command. To remove the access lists, use the **no** form of this command.

```
access-list access-list-number {deny | permit} protocol source source-wildcard destination
destination-wildcard [precedence precedence] [tos tos]
no access-list access-list-number
```

For ICMP, you can also use the following syntax:

```
access-list access-list-number {deny | permit} icmp source source-wildcard destination
destination-wildcard [icmp-type [icmp-code] | icmp-message] [precedence precedence]
[tos tos]
```

For IGMP, you can also use the following syntax:

```
access-list access-list-number {deny | permit} igmp source source-wildcard destination
destination-wildcard [igmp-type] [precedence precedence] [tos tos]
```

For TCP, you can also use the following syntax:

```
access-list access-list-number {deny | permit} tcp source source-wildcard
[operator port [port]] destination destination-wildcard
[operator port [port]] [established] [precedence precedence] [tos tos]
```

For UDP, you can also use the following syntax:

```
access-list access-list-number {deny | permit} udp source source-wildcard
[operator port [port]] destination destination-wildcard
[operator port [port]] [precedence precedence] [tos tos]
```



Caution Enhancements to this command are backward compatible; migrating from existing releases to Cisco IOS Release 10.3 will convert your access lists automatically. However, releases prior to Cisco IOS Release 10.3 are not upwardly compatible with these enhancements. Therefore, if you save an access list with these images and then use software prior to Cisco IOS Release 10.3, the resulting access list will not be interpreted correctly. *This could cause you severe security problems.* Save your old configuration file before booting these images.

Syntax Description

<i>access-list-number</i>	Number of an access list. This is a decimal number from 100 through 199.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
<i>protocol</i>	Name or number of an IP protocol. It can be one of the keywords eigrp , gre , icmp , igmp , igrp , ip , ipinip , nos , ospf , tcp , or udp , or an integer in the range 0 through 255 representing an IP protocol number. To match any Internet protocol, including ICMP, TCP, and UDP, use the keyword ip . Some protocols allow further qualifiers described below.

<i>source</i>	<p>Number of the network or host from which the packet is being sent. There are three alternative ways to specify the source:</p> <ul style="list-style-type: none">• Use a 32-bit quantity in four-part dotted-decimal format.• Use the keyword any as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255.• Use host source as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of <i>source</i> 0.0.0.0.
<i>source-wildcard</i>	<p>Wildcard bits to be applied to source. There are three alternative ways to specify the source wildcard:</p> <ul style="list-style-type: none">• Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore.• Use the keyword any as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255.• Use host source as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of <i>source</i> 0.0.0.0.
<i>destination</i>	<p>Number of the network or host to which the packet is being sent. There are three alternative ways to specify the destination:</p> <ul style="list-style-type: none">• Use a 32-bit quantity in four-part dotted-decimal format.• Use the keyword any as an abbreviation for the <i>destination</i> and <i>destination-wildcard</i> of 0.0.0.0 255.255.255.255.• Use host destination as an abbreviation for a <i>destination</i> and <i>destination-wildcard</i> of <i>destination</i> 0.0.0.0.
<i>destination-wildcard</i>	<p>Wildcard bits to be applied to the destination. There are three alternative ways to specify the destination wildcard:</p> <ul style="list-style-type: none">• Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore.• Use the keyword any as an abbreviation for a <i>destination</i> and <i>destination-wildcard</i> of 0.0.0.0 255.255.255.255.• Use host destination as an abbreviation for a <i>destination</i> and <i>destination-wildcard</i> of <i>destination</i> 0.0.0.0.
precedence <i>precedence</i>	<p>(Optional) Packets can be filtered by precedence level, as specified by a number from 0 to 7 or by name as listed in Table 17-1 in the Usage Guidelines section for this command.</p>
tos <i>tos</i>	<p>(Optional) Packets can be filtered by type of service level, as specified by a number from 0 to 15 or by name as listed in Table 17-1 in the Usage Guidelines section for this command.</p>
<i>icmp-type</i>	<p>(Optional) ICMP packets can be filtered by ICMP message type. The type is a number from 0 to 255.</p>
<i>icmp-code</i>	<p>(Optional) ICMP packets which are filtered by ICMP message type can also be filtered by the ICMP message code. The code is a number from 0 to 255.</p>

<i>icmp-message</i>	(Optional) ICMP packets can be filtered by an ICMP message type name or ICMP message type and code name. The possible names are found in the Usage Guidelines section for this command.
<i>igmp-type</i>	(Optional) IGMP packets can be filtered by IGMP message type or message name. A message type is a number from 0 to 15. IGMP message names are listed in the Usage Guidelines section for this command.
<i>operator</i>	(Optional) Compares source or destination ports. Possible operands include lt (less than), gt (greater than), eq (equal), neq (not equal), and range (inclusive range). If the operator is positioned after the <i>source</i> and <i>source-wildcard</i> , it must match the source port. If the operator is positioned after the <i>destination</i> and <i>destination-wildcard</i> , it must match the destination port. The range operator requires two port numbers. All other operators require one port number.
<i>port</i>	(Optional) The decimal number or name of a TCP or UDP port. A port number is a number from 0 to 65535. TCP port names are listed in the section “Usage Guidelines.” TCP port names can only be used when filtering TCP. UDP port names are listed in the section “Usage Guidelines.” UDP port names can only be used when filtering UDP.
established	(Optional) For the TCP protocol only: Indicates an established connection. A match occurs if the TCP datagram has the ACK or RST bits set. The nonmatching case enables the initial TCP datagram to form a connection.

Default

An extended access list defaults to a list that denies everything. An extended access list is terminated by an implicit deny statement.

Command Mode

Global configuration

Usage Guidelines

You can use access lists to control the transmission of packets on an interface, control virtual terminal line access, and restrict contents of routing updates. The communication server stops checking the extended access list after a match occurs.

Fragmented IP packets, other than the initial fragment, are immediately accepted by any extended IP access list. Extended access lists used to control virtual terminal line access or restrict contents of routing updates must not match against the TCP source port, the type of service value, or the packet’s precedence.

Note After an access list is created initially, any subsequent additions (possibly entered from the terminal) are placed at the end of the list. In other words, you cannot selectively add or remove access list command lines from a specific access list.

The following is a list of precedence names.

- **critical**
- **flash**
- **flash-override**
- **immediate**
- **internet**
- **network**
- **priority**
- **routine**

The following is a list of type of service (tos) names.

- **max-reliability**
- **max-throughput**
- **min-delay**
- **min-monetary-cost**
- **normal**

The following is a list of ICMP message type names and ICMP message type and code names.

- **administratively-prohibited**
- **alternate-address**
- **conversion-error**
- **dod-host-prohibited**
- **dod-net-prohibited**
- **echo**
- **echo-reply**
- **general-parameter-problem**
- **host-isolated**
- **host-precedence-unreachable**
- **host-redirect**
- **host-tos-redirect**
- **host-tos-unreachable**
- **host-unknown**
- **host-unreachable**
- **information-reply**

- **information-request**
- **mask-reply**
- **mask-request**
- **mobile-redirect**
- **net-redirect**
- **net-tos-redirect**
- **net-tos-unreachable**
- **net-unreachable**
- **network-unknown**
- **no-room-for-option**
- **option-missing**
- **packet-too-big**
- **parameter-problem**
- **port-unreachable**
- **precedence-unreachable**
- **protocol-unreachable**
- **reassembly-timeout**
- **redirect**
- **router-advertisement**
- **router-solicitation**
- **source-quench**
- **source-route-failed**
- **time-exceeded**
- **timestamp-reply**
- **timestamp-request**
- **traceroute**
- **ttl-exceeded**
- **unreachable**

The following is a list of IGMP message names.

- **dvmrp**
- **host-query**
- **host-report**
- **pim**
- **trace**

The following is a list of TCP port names that can be used instead of port numbers. Refer to the current Assigned Numbers RFC to find a reference to these protocols. Port numbers corresponding to these protocols can also be found by typing a ? in the place of a port number.

- **bgp**
- **chargen**
- **daytime**
- **discard**
- **domain**
- **echo**
- **finger**
- **ftp**
- **ftp-data**
- **gopher**
- **hostname**
- **irc**
- **klogin**
- **kshell**
- **lpd**
- **nntp**
- **pop2**
- **pop3**
- **smtp**
- **sunrpc**
- **syslog**
- **tacacs-ds**
- **talk**
- **telnet**
- **time**
- **uucp**
- **whois**
- **www**

The following is a list of UDP port names that can be used instead of port numbers. Refer to the current Assigned Numbers RFC to find a reference to these protocols. Port numbers that correspond to these protocols can also be found by typing a ? in the place of a port number.

- **biff**
- **bootpc**
- **bootps**
- **discard**
- **dns**
- **dnsix**

- **echo**
- **mobile-ip**
- **nameserver**
- **netbios-dgm**
- **netbios-ns**
- **ntp**
- **rip**
- **snmp**
- **snmptrap**
- **sunrpc**
- **syslog**
- **tacacs-ds**
- **talk**
- **tftp**
- **time**
- **who**
- **xdmcp**

The following is a list of UDP port names that can be used instead of port numbers. Refer to the current Assigned Numbers RFC to find a reference to these protocols. Port numbers that correspond to these protocols can also be found by typing a ? in the place of a port number.

- **biff**
- **bootpc**
- **bootps**
- **discard**
- **dns**
- **dnsix**
- **echo**
- **mobile-ip**
- **nameserver**
- **netbios-dgm**
- **netbios-ns**
- **ntp**
- **rip**
- **snmp**
- **snmptrap**
- **sunrpc**
- **syslog**

- **tacacs-ds**
- **talk**
- **tftp**
- **time**
- **who**
- **xdmcp**

Examples

In the following example, serial interface 0 is part of a Class B network with the address 128.88.0.0, and the mail host's address is 128.88.1.2. The keyword **established** is used only for the TCP protocol to indicate an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which indicate that the packet belongs to an existing connection.

```
access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.0.0 0.0.255.255 established
access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.1.2 0.0.0.0 eq 25
interface serial 0
ip access-group 102 in
```

The following example also permit DNS packets and ICMP echo and echo reply packets.

```
access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
access-list 102 permit tcp any host 128.88.1.2 eq smtp
access-list 102 permit tcp any any eq domain
access-list 102 permit udp any any eq domain
access-list 102 permit icmp any any echo
access-list 102 permit icmp any any echo-reply
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-class
access-list (standard)
distribute-list †
ip access-group
priority-list †
queue-list †
show access-lists
show ip access-list

arp (global)

To add a permanent entry in the ARP cache, use the **arp** global configuration command. To remove an entry from the ARP cache, use the **no** form of this command.

```
arp ip-address hardware-address type [alias]  
no arp ip-address hardware-address type [alias]
```

Syntax Description

<i>ip-address</i>	IP address in four-part dotted-decimal format corresponding to the local data link address.
<i>hardware-address</i>	Local data link address (a 48-bit address).
<i>type</i>	Encapsulation description. For Ethernet interfaces, this is typically the arpa keyword. For Token Ring interfaces, this is always snap .
alias	(Optional) Indicates that the communication server should respond to ARP requests as if it were the owner of the specified address.

Default

No entries are permanently installed in the ARP cache.

Command Mode

Global configuration

Usage Guidelines

The communication server uses ARP cache entries to translate 32-bit Internet Protocol addresses into 48-bit hardware addresses.

Because most hosts support dynamic resolution, you generally do not need to specify static ARP cache entries.

To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command.

Example

The following is an example of a static ARP entry for a typical Ethernet host:

```
arp 192.31.7.19 0800.0900.1834 arpa
```

Related Command

clear arp-cache

arp (interface)

To control the interface-specific handling of IP address resolution into 48-bit Ethernet, FDDI, and Token Ring hardware addresses, use the **arp** interface configuration command. To disable an encapsulation type, use the **no** form of this command.

```
arp {arpa | probe | snap}
no arp {arpa | probe | snap}
```

Syntax Description

arpa	Standard Ethernet-style ARP (RFC 826)
probe	HP Probe protocol for IEEE-802.3 networks
snap	ARP packets conforming to RFC 1042

Default

Standard Ethernet-style ARP

Command Mode

Interface configuration

Usage Guidelines

Unlike most commands that take multiple arguments, arguments to the **arp** command are not mutually exclusive. Each command enables or disables a specific type of ARP. For example, if you enter the **arp arpa** command followed by the **arp probe** command, the communication server would send three (two for **probe** and one for **arpa**) packets each time it needed to discover a MAC address.

The **arp probe** command allows the communication server to use the Probe protocol (in addition to ARP) whenever it attempts to resolve an IEEE-802.3 or Ethernet local data link address. The subset of Probe that performs address resolution is called Virtual Address Request and Reply. Using Probe, the communication server can communicate transparently with Hewlett-Packard IEEE-802.3 hosts that use this type of data encapsulation.

Note Cisco's support for HP Probe proxy support changed as of Software Release 8.3(2) and subsequent software releases. The **no arp probe** command is now the default. All interfaces that will use Probe must now be explicitly configured for **arp probe**.

The **show interfaces EXEC** command displays the type of ARP being used on a particular interface. To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command.

Example

The following example enables probe services:

```
interface ethernet 0
  arp probe
```

Related Commands

clear arp-cache

show interfaces

arp timeout

To configure how long an entry remains in the ARP cache, use the **arp timeout** interface configuration command. To restore the default value, use the **no** form of this command.

arp timeout *seconds*
no arp timeout *seconds*

Syntax Description

seconds Time, in seconds, that an entry remains in the ARP cache. A value of zero means that entries are never cleared from the cache.

Default

14400 seconds (4 hours)

Command Mode

Interface configuration

Usage Guidelines

This command is ignored when issued on interfaces that do not use ARP. The **show interfaces EXEC** command displays the ARP timeout value. The value follows the “Entry Timeout:” heading, as seen in this sample **show interfaces** display:

```
ARP type: ARPA, PROBE, Entry Timeout: 14400 sec
```

Example

The following example illustrates how to set the ARP timeout to 12000 seconds to allow entries to time out more quickly than the default:

```
interface ethernet 0
  arp timeout 12000
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show interfaces †

clear arp-cache

To delete all dynamic entries from the ARP cache, to clear the fast-switching cache, and to clear the IP route cache, use the **clear arp-cache** EXEC command.

clear arp-cache

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Example

The following example removes all dynamic entries from the ARP cache and clears the fast-switching cache:

```
clear arp-cache
```

Related Commands

arp (global)

arp (interface)

clear host

To delete entries from the host-name-and-address cache, use the **clear host** EXEC command.

```
clear host {name | *}
```

Syntax Description

name Particular host entry to remove.

* Removes all entries.

Command Mode

EXEC

Usage Guidelines

The host name entries will not be removed from NVRAM, but will be cleared in running memory.

Example

The following example clears all entries from the host name-and-address cache:

```
clear host *
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ip host

show hosts †

clear ip accounting

To clear the active or checkpointed database when IP accounting is enabled, use the **clear ip accounting** EXEC command.

clear ip accounting [checkpoint]

Syntax Description

checkpoint (Optional) Clears the checkpointed database

Command Mode

EXEC

Usage Guidelines

You can also clear the checkpointed database by issuing the **clear ip accounting** command twice in succession.

Example

The following example clears the active database when IP accounting is enabled:

```
clear ip accounting
```

Related Commands

ip accounting

ip accounting-list

ip accounting-threshold

ip accounting-transits

show ip accounting

clear ip nhrp

To clear all dynamic entries from the Next Hop Resolution Protocol (NHRP) cache, use the **clear ip nhrp** EXEC command.

clear ip nhrp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command does not clear any static (configured) IP-to-NBMA address mappings from the NHRP cache.

Example

In the following example, all dynamic entries are cleared from the NHRP cache for the interface:

```
clear ip nhrp
```

Related Command

show ip nhrp

clear ip route

To delete routes from the IP routing table, use the **clear ip route** EXEC command.

clear ip route {*network* [*mask*] | *}

Syntax Description

<i>network</i>	Network or subnet address to remove.
<i>mask</i>	(Optional) Subnet address to remove.
*	Removes all routing table entries.

Default

All entries are removed.

Command Mode

EXEC

Example

The following example removes a route to network 132.5.0.0 from the IP routing table:

```
clear ip route 132.5.0.0
```

dnsix-dmdp retries

To set the retransmit count used by the DNSIX Message Delivery Protocol (DMDP), use the **dnsix-dmdp retries** global configuration command. To restore the default number of retries, use the **no** form of this command.

```
dnsix-dmdp retries count  
no dnsix-dmdp retries count
```

Syntax Description

count Number of times DMDP will retransmit a message. It can be a decimal integer from 0 through 200. The default is 4 retries, or until acknowledged.

Default

Retransmits messages up to 4 times, or until acknowledged.

Command Mode

Global configuration

Example

The following example sets the number of times DMDP will attempt to retransmit a message to 150:

```
dnsix-dmdp retries 150
```

Related Commands

```
dnsix-nat authorized-redirect  
dnsix-nat primary  
dnsix-nat secondary  
dnsix-nat source  
dnsix-nat transmit-count
```

dnsix-nat authorized-redirect

To specify the address of a collection center that is authorized to change the primary and secondary addresses of the host to receive audit messages, use the **dnsix-nat authorized-redirect** global configuration command. To delete an address, use the **no** form of this command.

dnsix-nat authorized-redirect *ip-address*
no dnsix-nat authorized-redirect *ip-address*

Syntax Description

<i>ip-address</i>	IP address of the host from which redirection requests are permitted
-------------------	--

Default

An empty list of addresses.

Command Mode

Global configuration

Usage Guidelines

Use multiple **dnsix-nat authorized-redirect** commands to specify a set of hosts that are authorized to change the destination for audit messages. Redirection requests are checked against the configured list, and if the address is not authorized the request is rejected and an audit message is generated. If no address is specified, no redirection messages are accepted.

Example

The following example specifies that the address of the collection center that is authorized to change the primary and secondary addresses is 193.1.1.1.

```
dnsix-nat authorization-redirect 193.1.1.1.
```


dnsix-nat primary

To specify the IP address of the host to which DNSIX audit messages are sent, use the **dnsix-nat primary** global configuration command. To delete an entry, use the **no** form of this command.

dnsix-nat primary *ip-address*
no dnsix-nat primary *ip-address*

Syntax Description

ip-address IP address for the primary collection center

Default

Messages are not sent.

Command Mode

Global configuration

Usage Guidelines

An IP address must be configured before audit messages can be sent.

Example

The following example configures an IP address as the address of the host to which DNSIX audit messages are sent:

```
dnsix-nat primary 194.1.1.1
```

dnsix-nat secondary

To specify an alternate IP address for the host to which DNSIX audit messages are sent, use the **dnsix-nat secondary** global configuration command. To delete an entry, use the **no** form of this command.

dnsix-nat secondary *ip-address*
no dnsix-nat secondary *ip-address*

Syntax Description

ip-address IP address for the secondary collection center

Default

No alternate IP address is known.

Command Mode

Global configuration

Usage Guidelines

When the primary collection center is unreachable, audit messages are sent to the secondary collection center instead.

Example

The following example configures an IP address as the address of an alternate host to which DNSIX audit messages are sent:

```
dnsix-nat secondary 193.1.1.1
```

dnsix-nat source

To start the audit-writing module and to define audit trail source address, use the **dnsix-nat source** global configuration command. To disable the DNSIX audit trail writing module, use the **no** form of this command.

```
dnsix-nat source ip-address  
no dnsix-nat source ip-address
```

Syntax Description

ip-address Source IP address for DNSIX audit messages

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

You must issue the **dnsix-nat source** command before any of the other **dnsix-nat** commands. The configured IP address is used as the source IP address for DMDP protocol packets sent to any of the collection centers.

Example

The following example enables the audit trail writing module, and specifies that the source IP address for any generated audit messages should be the same as the primary IP address of Ethernet interface 0.

```
dnsix-nat source 128.105.2.5  
interface ethernet 0  
ip address 128.105.2.5 255.255.255.0
```

dnsix-nat transmit-count

To have the audit writing module collect multiple audit messages in the buffer before sending the messages to a collection center, use the **dnsix-nat transmit-count** global configuration command. To revert to the default audit message count, use the **no** form of this command.

dnsix-nat transmit-count *count*
no dnsix-nat transmit-count *count*

Syntax Description

<i>count</i>	Number of audit messages to buffer before transmitting to the server. Integer from 1 through 200.
--------------	---

Default

One message is sent at a time.

Command Mode

Global configuration

Usage Guidelines

An audit message is sent as soon as the message is generated by the IP packet-processing code. The audit writing module can, instead, buffer up to several audit messages before transmitting to a collection center.

Example

The following example configures the system to buffer five audit messages before transmitting them to a collection center:

```
dnsix-nat transmit-count 5
```

ip access-group

To control access to an interface, use the **ip access-group** interface configuration command. To remove the specified access group, use the **no** form of this command.

```
ip access-group access-list-number { in | out }  
no ip access-group access-list-number { in | out }
```

Syntax Description

<i>access-list-number</i>	Number of an access list. This is a decimal number from 1 through 199.
in	Filters on inbound packets.
out	Filters on outbound packets.

Default

Entering a keyword is strongly recommended, but if a keyword is not specified, **out** is the default.

Command Mode

Interface configuration

Usage Guidelines

For inbound access lists, after receiving a packet, the communication server checks the source address of the packet against the access list. If the access list permits the address, the communication server continues to process the packet. If the access list rejects the address, the communication server discards the packet and returns an ICMP *Host Unreachable* message.

For outbound access lists, after receiving and routing a packet to a controlled interface, the communication server checks the source address of the packet against the access list. If the access list permits the address, the communication server transmits the packet. If the access list rejects the address, the communication server discards the packet and returns an ICMP Host Unreachable message.

Access lists are applied on either outbound or inbound interfaces.

If the specified access list does not exist, all packets are passed.

Example

The following example applies list 101 on packets outbound from Ethernet interface 0:

```
interface ethernet 0  
ip access-group 101 out
```

Related Commands

access-list (extended)
show access-lists

ip accounting

To enable IP accounting on an interface, use the **ip accounting** interface configuration command. To disable IP accounting, use the **no** form of this command.

```
ip accounting [access-violations]  
no ip accounting [access-violations]
```

Syntax Description

access-violations (Optional) Enables IP accounting with the ability to identify IP traffic that fails IP access lists.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP accounting records the number of bytes (IP header and data) and IP packets switched through the communication server for each source and destination pair. Only transit IP traffic is measured and only on an outbound basis; traffic generated by the communication server or terminating in the communication server is not included in the accounting statistics.

If you specify the **access-violations** keyword, this command provides information identifying IP traffic that fails IP access lists. Identifying IP source addresses that violate IP access lists alerts you to possible attempts to breach security. The data might also indicate that you should verify IP access list configurations.

Statistics are accurate even if IP fast switching or IP access lists are being used on the interface.

IP accounting disables autonomous switching and SSE switching on the interface.

Example

The following example enables IP accounting on Ethernet interface 0:

```
interface ethernet 0  
ip accounting
```

Related Commands

```
clear ip accounting  
ip accounting-list  
ip accounting-threshold  
ip accounting-transits  
show ip accounting
```

ip accounting-list

To define filters to control the hosts for which IP accounting information is kept, use the **ip accounting-list** global configuration command. To remove a filter definition, use the **no** form of this command.

```
ip accounting-list ip-address mask  
no ip accounting-list ip-address mask
```

Syntax Description

<i>ip-address</i>	IP address in dotted-decimal format
<i>mask</i>	IP mask

Default

No filters are defined.

Command Mode

Global configuration

Usage Guidelines

The source and destination address of each IP datagram is logically ANDed with the *mask* and compared with the *ip-address*. If there is a match, the information about the IP datagram will be entered into the accounting database. If there is no match, the IP datagram is considered a *transit* datagram and will be counted according to the setting of the **ip accounting-transits** global configuration command.

Example

The following example adds all hosts with IP addresses beginning with 192.31 to the list of hosts for which accounting information will be kept:

```
ip accounting-list 192.31.0.0 255.255.0.0
```

Related Commands

```
clear ip accounting  
ip accounting  
ip accounting-threshold  
ip accounting-transits  
show ip accounting
```

ip accounting-threshold

To set the maximum number of accounting entries to be created, use the **ip accounting-threshold** global configuration command. To restore the default number of entries, use the **no** form of this command.

ip accounting-threshold *threshold*
no ip accounting-threshold *threshold*

Syntax Description

threshold Maximum number of entries (source and destination address pairs) that the communication server accumulates.

Default

512 entries

Command Mode

Global configuration

Usage Guidelines

The accounting threshold defines the maximum number of entries (source and destination address pairs) that the communication server accumulates, preventing IP accounting from possibly consuming all available free memory. This level of memory consumption could occur in a communication server that is switching traffic for many hosts. Overflows will be recorded; see the monitoring commands for display formats.

The default accounting threshold of 512 entries results in a maximum table size of 12928 bytes. Active and checkpointed tables can reach this size independently.

Example

The following example sets the IP accounting threshold to only 500 entries:

```
ip accounting-threshold 500
```

Related Commands

clear ip accounting
ip accounting
ip accounting-list
ip accounting-transits
show ip accounting

ip accounting-transits

To control the number of transit records that are stored in the IP accounting database, use the **ip accounting-transits** global configuration command. To return to the default number of records, use the **no** form of this command.

```
ip accounting-transits count  
no ip accounting-transits
```

Syntax Description

count Number of transit records to store in the IP accounting database

Default

0

Command Mode

Global configuration

Usage Guidelines

Transit entries are those that do not match any of the filters specified by **ip accounting-list** global configuration commands. If no filters are defined, no transit entries are possible.

To maintain accurate accounting totals, the communication server software maintains two accounting databases: an active and a checkpointed database.

Example

The following example specifies that no more than 100 transit records are stored:

```
ip accounting-transits 100
```

Related Commands

```
clear ip accounting  
ip accounting  
ip accounting-list  
ip accounting-threshold  
show ip accounting
```

ip address

To set an IP address for an interface, use the **ip address** interface configuration command. To remove an IP address, use the **no** form of this command.

```
ip address ip-address mask  
no ip address ip-address mask
```

Syntax Description

<i>ip-address</i>	IP address
<i>mask</i>	Mask for the associated IP subnet

Default

No IP address is defined for an interface.

Command Mode

Interface configuration

Usage Guidelines

Hosts can determine subnet masks using the Internet Control Message Protocol (ICMP) Mask Request message. Communication servers respond to this request with an ICMP Mask Reply message.

You can disable IP processing on a particular interface by removing its IP address with the **no ip address** command. If the communication server detects another host using one of its IP addresses, it will print an error message on the console.

Example

In the following example, 131.108.1.27 is the primary address for Ethernet interface 0:

```
interface ethernet 0  
ip address 131.108.1.27 255.255.255.0
```

ip address secondary

To set multiple IP addresses for an interface, use the **ip address secondary** interface configuration command. To remove an address, use the **no** form of this command.

```
ip address ip-address mask secondary  
no ip address ip-address mask secondary
```

Syntax Description]

<i>ip-address</i>	IP address
<i>mask</i>	Mask for the associated IP subnet

Default

No secondary IP addresses are defined.

Command Mode

Interface configuration

Usage Guidelines

Hosts can determine subnet masks using the Internet Control Message Protocol (ICMP) Mask Request message. Communication servers respond to this request with an ICMP Mask Reply message.

Packets generated by the communication server always use the primary interface IP address. Therefore, all communication servers on a segment should share the same primary network number.

Note When you are routing OSPF, ensure that all secondary addresses of an interface fall into the same OSPF area as the primary addresses.

Example

In the following example, 131.108.1.27 is the primary address and 192.31.7.17 and 192.31.8.17 are secondary addresses for Ethernet interface 0:

```
interface ethernet 0  
ip address 131.108.1.27 255.255.255.0  
ip address 192.31.7.17 255.255.255.0 secondary  
ip address 192.31.8.17 255.255.255.0 secondary
```

ip broadcast-address

To define a broadcast address for an interface, use the **ip broadcast-address** interface configuration command. To restore the default IP broadcast address, use the **no** form of this command.

```
ip broadcast-address [ip-address]  
no ip broadcast-address [ip-address]
```

Syntax Description

ip-address (Optional) IP broadcast address for a network

Default

Default address: 255.255.255.255 (all ones)

Command Mode

Interface configuration

Example

The following example specifies an IP broadcast address of 0.0.0.0:

```
ip broadcast-address 0.0.0.0
```

ip cache-invalidate-delay

To control the invalidation rate of the IP route cache, use the **ip cache-invalidate-delay** global configuration command. To allow the IP route cache to be invalidated immediately, use the **no** form of this command.

```
ip cache-invalidate-delay [minimum maximum quiet threshold]  
no ip cache-invalidate-delay
```

Syntax Description

<i>minimum</i>	(Optional) Minimum time, in seconds, between invalidation request and actual invalidation. The default is 2 seconds.
<i>maximum</i>	(Optional) Maximum time, in seconds, between invalidation request and actual invalidation. The default is 5 seconds.
<i>quiet</i>	(Optional) Length of quiet period, in seconds, before invalidation.
<i>threshold</i>	(Optional) Maximum number of invalidation requests considered to be quiet.

Default

minimum = 2 seconds

maximum = 5 seconds, and 3 seconds with no more than zero invalidation requests

Command Mode

Global configuration

Usage Guidelines

All cache invalidation requests are honored immediately.

This command should typically not be used except under the guidance of technical support personnel. Incorrect settings can seriously degrade network performance.

The IP fast switching feature maintains a cache of IP routes for rapid access. When a packet is to be forwarded and the corresponding route is not present in the cache, the packet is process-switched and a new cache entry is built. However, when routing table changes occur (such as when a link or an interface goes down), the route cache must be flushed so that it can be rebuilt with up-to-date routing information.

This command controls how the route cache is flushed. The intent is to delay invalidation of the cache until after routing has settled down, because there tend to be many route table changes clustered in a short period of time, and the cache may be flushed repeatedly, which may put a high CPU load on the router.

When this feature is enabled, and the system requests that the route cache be flushed, the request is held for at least *minimum* seconds. Then the system determines whether the cache has been “quiet,” that is, less than *threshold* invalidation requests in the last *quiet* seconds. If the cache has been quiet, the cache is then flushed. If the cache does not become quiet within *maximum* seconds after the first request, it is flushed unconditionally.

Manipulation of these parameters trades off CPU utilization versus route convergence time. Note that this does not affect the timing of the routing protocols, but only of the removal of stale cache entries.

Example

The following example sets a minimum delay of 5 seconds, a maximum delay of 30 seconds, and a quiet threshold of no more than 5 invalidation requests in the previous 10 seconds:

```
ip cache-invalidate-delay 5 30 10 5
```

Related Commands

ip route-cache
show ip cache

ip classless

At times the router might receive packets destined for a subnet of a network that has no network default route. To have the router forward such packets to the best supernet route possible, use the **ip classless** global configuration command. To disable this feature, use the **no** form of this command.

ip classless
no ip classless

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

This command allows the router to forward packets that are destined for unrecognized subnets of directly connected networks. By default, when a router receives packets for a subnet that numerically falls within its subnetwork addressing scheme, if there is no such subnet number in the routing table and there is no network default route, the router discards the packets. However, when the **ip classless** command is enabled, the router instead forwards those packets to the best supernet route.

Example

The following example configures the router to forward packets destined for an unrecognized subnet to the best supernet possible:

```
ip classless
```

ip default-gateway

To define a default gateway (router) when IP routing is disabled, use the **ip default-gateway** global configuration command. To disable this function, use the no form of this command.

ip default-gateway *ip-address*
no ip default-gateway *ip-address*

Syntax Description

ip-address IP address of the communication server

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The host sends any packets that need the assistance of a gateway to the address you specify. If another gateway has a better route to the requested host, the default gateway sends an ICMP redirect message to the server. The ICMP redirect message indicates which local communication server the server should use.

Example

The following example defines the communication server on IP address 192.31.7.18 as the default communication server:

```
ip default-gateway 192.31.7.18
```

Related Command

show ip redirects

ip directed-broadcast

To enable the translation of directed broadcast to physical broadcasts, use the **ip directed-broadcast** interface configuration command. To disable this function, use the **no** form of this command.

```
ip directed-broadcast [access-list-number]  
no ip directed-broadcast [access-list-number]
```

Syntax Description

access-list-number (Optional) Number of the access list. If specified, a broadcast must pass the access list to be forwarded. If not specified, all broadcasts are forwarded.

Default

Enabled, with no list specified.

Command Mode

Interface configuration

Usage Guidelines

This feature is enabled only for those protocols configured using the **ip forward-protocol** global configuration command. An access list may be specified to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts.

Example

The following example enables forwarding of IP directed broadcasts on Ethernet interface 0:

```
interface ethernet 0  
ip directed-broadcast
```

Related Command

ip forward-protocol

ip domain-list

To define a list of default domain names to complete unqualified host names, use the **ip domain-list** global configuration command. To delete a name from a list, use the **no** form of this command.

ip domain-list *name*
no ip domain-list *name*

Syntax Description

name Domain name. Do not include the initial period that separates an unqualified name from the domain name.

Default

No domain names are defined.

Command Mode

Global configuration

Usage Guidelines

If there is no domain list, the domain name that you specified with the **ip domain-name** global configuration command is used. If there is a domain list, the default domain name is not used. The **ip domain-list** command is similar to the **ip domain-name** command, except that with **ip domain-list** you can define a list of domains, each to be tried in turn.

Examples

The following example adds several domain names to a list:

```
ip domain-list martinez.com  
ip domain-list stanford.edu
```

The following example adds a name to and then deletes a name from the list:

```
ip domain-list sunya.edu  
no ip domain-list stanford.edu
```

Related Command

ip domain-name

ip domain-lookup

To enable the IP Domain Name System-based host name-to-address translation, use the **ip domain-lookup** global configuration command. To disable the Domain Name System, use the **no** form of this command.

```
ip domain-lookup  
no ip domain-lookup
```

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Example

The following example enables the IP Domain Name System-based host name-to-address translation:

```
ip domain-lookup
```

Related Commands

```
ip domain-lookup nsap  
ip domain-name  
ip name-server
```

ip domain-name

To define a default domain name that the communication server uses to complete unqualified host names (names without a dotted-decimal domain name), use the **ip domain-name** global configuration command. To disable use of the Domain Name System, use the **no** form of this command.

ip domain-name *name*
no ip domain-name

Syntax Description

name Default domain name used to complete unqualified host names. Do not include the initial period that separates an unqualified name from the domain name.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

Any IP host name that does not contain a domain name (that is, any name without a dot), will have the dot and cisco.com appended to it before being added to the host table.

Example

The following example defines cisco.com as the default domain name:

```
ip domain-name cisco.com
```

Related Commands

ip domain-list
ip domain-lookup
ip name-server

ip forward-protocol

To specify which protocols and ports the communication server forwards when forwarding broadcast packets, use the **ip forward-protocol** global configuration command. To remove a protocol or port, use the **no** form of this command.

```
ip forward-protocol { udp [port] | nd | sdns }
no ip forward-protocol { udp [port] | nd | sdns }
```

Syntax Description

udp	Forward User Datagram Protocol (UDP) datagrams. See the “Default” section for this command for a list of port numbers forwarded by default.
<i>port</i>	(Optional) Destination port that controls which UDP services are forwarded.
nd	Forward Network Disk (ND) datagrams. This protocol is used by older diskless SUN workstations.
sdns	Secure Data Network Service.

Default

If an IP helper address is defined, UDP forwarding is enabled on the default ports. If UDP flooding is configured, UDP flooding is enabled on the default ports.

If a helper address is specified and UDP forwarding is enabled, broadcast packets destined to the following port numbers are forwarded by default:

- Trivial File Transfer (TFTP) (port 69)
- Domain Name System (port 53)
- Time service (port 37)
- NetBIOS Name Server (port 137)
- NetBIOS Datagram Server (port 138)
- Boot Protocol (BOOTP) client and server datagrams (ports 67 and 68)
- Regular TACACS service (port 49)

Note Using the **ip directed-broadcast** interface configuration command with the optional *access-list-number* argument overrides the **ip forward-protocol** command.

Command Mode

Global configuration

Usage Guidelines

Enabling a helper address or UDP flooding on an interface causes the communication server to forward particular broadcast packets. You can use **ip forward-protocol** command to specify exactly which types of broadcast packets you would like to have forwarded. A number of commonly forwarded applications are enabled by default. Enabling forwarding for some ports (for example, RIP) may be hazardous to your network.

For example, to enable forwarding and flooding on the default ports, use the **ip forward-protocol** command, specifying just UDP without the port.

One common application that requires helper addresses is Dynamic Host Configuration Protocol (DHCP). DHCP is defined in RFC 1531. DHCP protocol information is carried inside of BOOTP packets. To enable BOOTP broadcast forwarding for a set of clients, configure a helper address on the communication server interface closest to the client. The helper address should specify the address of the DHCP server. If you have multiple servers, you can configure one helper address for each server. Because BOOTP packets are forwarded by default, DHCP information can now be forwarded by the communication server. The DHCP server now receives broadcasts from the DHCP clients.

Example

The following example uses the **ip forward-protocol** command to specify forwarding of UDP port 3001 in addition to the default ports, and then defines a helper address:

```
ip forward-protocol udp 3001
!
interface ethernet 1
ip helper-address 131.120.1.0
```

Related Commands

ip directed-broadcast
ip forward-protocol spanning-tree
ip forward-protocol turbo-flood
ip helper-address

ip forward-protocol any-local-broadcast

To forward any broadcasts including local subnet broadcasts, use the **ip forward-protocol any-local-broadcast** global configuration command. To disable this type of forwarding, use the **no** form of this command.

```
ip forward-protocol any-local-broadcast  
no ip forward-protocol any-local-broadcast
```

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **ip forward-protocol any-local-broadcast** command forwards packets similarly to the way **ip forward-protocol spanning-tree** command does. That is, it forwards packets whose contents are all ones (255.255.255.255), all zeros (0.0.0.0), and, if subnetting is enabled, all networks (for example 131.108.255.255 in the network number 131.108.0.0). This mechanism also forwards packets whose contents are the zeros version of the all-networks broadcast when subnetting is enabled (for example, 131.108.0.0). In addition, it forwards any local subnet broadcast packets.

Use the **ip forward-protocol any-local-broadcast** command in conjunction with the **ip forward-protocol spanning-tree** command, not as a replacement for it.

Example

Assume that a communication server is directly connected to subnet 1 of network 131.108.0.0 and that the network mask is 255.255.255.0. 1.0 . In addition to the broadcast addresses mentioned in the “Usage Guidelines” section, the following command enables the forwarding of IP broadcasts destined to 131.108.1.255 and 131.108:

```
ip forward-protocol any-local-broadcast
```

Related Command

ip forward-protocol spanning-tree

ip gdp gdp

To configure the router discovery feature using the Cisco Gateway Discovery Protocol (GDP) routing protocol, use the **ip gdp gdp** interface configuration command. To disable this feature, use the **no** form of this command.

```
ip gdp gdp  
no ip gdp gdp
```

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures router discovery using GDP on Ethernet interface 0:

```
interface ethernet 0  
ip gdp gdp
```


ip gdp igrp

To configure the router discovery feature using the Cisco Interior Gateway Routing Protocol (IGRP), use the **ip gdp igrp** interface configuration command. To disable this feature, use the **no** form of this command.

```
ip gdp igrp  
no ip gdp igrp
```

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures communication server discovery using IGRP on Ethernet interface 1:

```
interface ethernet 1  
ip gdp igrp
```

ip gdp irdp

To configure the router discovery feature using the ICMP Router Discovery Protocol (IRDP), use the **ip gdp irdp** interface configuration command. To disable this feature, use the **no** form of this command.

ip gdp irdp
no ip gdp irdp

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures communication server discovery using IRDP on the Ethernet interface 0:

```
interface ethernet 0
ip gdp irdp
```

ip gdp rip

To configure the router discovery feature using the Routing Information Protocol (RIP), use the **ip gdp rip** interface configuration command. To disable this feature, use the **no** form of this command.

```
ip gdp rip  
no ip gdp rip
```

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures communication server discovery using RIP on Ethernet interface 1:

```
interface ethernet 1  
ip gdp rip
```

ip helper-address

To have the communication server forward User Datagram Protocol (UDP) broadcasts, including BOOTP, received on an interface, use the **ip helper-address** interface configuration command. To disable the forwarding of broadcast packets to specific addresses, use the **no** form of this command.

ip helper-address *address*
no ip helper-address *address*

Syntax Description

address Destination broadcast or host address to be used when forwarding UDP broadcasts. You can have more than one helper address per interface.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Combined with the **ip forward-protocol** global configuration command, the **ip helper-address** command allows you to control which broadcast packets and which protocols are forwarded.

One common application which requires helper addresses is Dynamic Host Configuration Protocol (DHCP). DHCP is defined in RFC 1531. DHCP protocol information is carried inside of BOOTP packets. To enable BOOTP broadcast forwarding for a set of clients, configure a helper address on the communication server interface closest to the client. The helper address should specify the address of the DHCP server. If you have multiple servers, you can configure one helper address for each server. Since BOOTP packets are forwarded by default, DHCP information can now be forwarded by the communication server. The DHCP server now receives broadcasts from the DHCP clients.

Example

The following example defines an address that acts as a helper address:

```
interface ethernet 1
 ip helper-address 121.24.43.2
```

Related Command

ip forward-protocol

ip host

To define a static host name-to-address mapping in the host cache, use the **ip host** global configuration command. To remove the name-to-address mapping, use the **no** form of this command.

```
ip host hostname [tcp-port-number] address1 [address2...address8]  
no ip host hostname address
```

Syntax Description

<i>hostname</i>	Name of the host. The first character can be either a letter or a number, but if you use a number, the operations you can perform are limited.
<i>tcp-port-number</i>	(Optional) TCP port number to connect to when using the defined host name in conjunction with an EXEC connect or telnet command. The default is Telnet (port 23).
<i>address</i>	Associated IP address. You can bind up to eight addresses to a host name.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The first character can be either a letter or a number, but if you use a number, the operations you can perform (such as ping) are limited.

Example

The following example uses the **ip host** command to define two static mappings:

```
ip host croff 192.31.7.18  
ip host bisso-gw 10.2.0.2 192.31.7.33
```

ip hp-host

To enter into the host table the host name of an HP host to be used for HP Probe Proxy service, use the **ip hp-host** global configuration command. To remove a host name, use the **no** form of this command.

ip hp-host *hostname ip-address*
no ip hp-host *hostname ip-address*

Syntax Description

<i>hostname</i>	Name of the host
<i>ip-address</i>	IP address of the host

Default

No host names are defined.

Command Mode

Global configuration

Usage Guidelines

To use the HP Proxy service, you must first enter the host name of the HP host into the host table using this command.

Example

The following example specifies an HP host's name and address, and then enables Probe Proxy:

```
ip hp-host BCWjo 131.108.1.27
interface ethernet 0
ip probe proxy
```

Related Command

ip probe proxy

ip mask-reply

To have the communication server to respond to Internet Control Message Protocol (ICMP) mask requests by sending ICMP Mask Reply messages, use the **ip mask-reply** interface configuration command. To disable this function, use the **no** form of this command.

ip mask-reply
no ip mask-reply

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Example

The following example enables the sending of ICMP Mask Reply messages on Ethernet interface 0:

```
interface ethernet 0
ip address 131.108.1.0 255.255.255.0
ip mask-reply
```

ip mtu

To set the maximum transmission unit (MTU) size of IP packets sent on an interface, use the **ip mtu** interface configuration command. To restore the default MTU size, use the **no** form of this command.

ip mtu *bytes*
no ip mtu

Syntax Description

bytes MTU in bytes

Default

Minimum is 128 bytes; maximum depends on interface medium

Command Mode

Interface configuration

Usage Guidelines

If an IP packet exceeds the MTU set for the communication server's interface, the communication server will fragment it.

All devices on a physical medium must have the same protocol MTU in order to operate.

Note Changing the MTU value (with the **mtu** interface configuration command) can affect the IP MTU value. If the current IP MTU value is the same as the MTU value, and you change the MTU value, the IP MTU value will be modified automatically to match the new MTU. However, the reverse is not true; changing the IP MTU value has no effect on the value for the **mtu** command.

Example

The following example sets the maximum IP packet size for the first serial interface to 300 bytes:

```
interface serial 0
ip mtu 300
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

mtu †

ip name-server

To specify the address of one or more name servers to use for name and address resolution, use the **ip name-server** global configuration command. To remove the addresses specified, use the **no** form of this command.

```
ip name-server server-address1 [[server-address2]... server-address6]  
no ip name-server server-address1 [[server-address2]... server-address6]
```

Syntax Description

server-address1...6 IP addresses of up to six name servers

Default

No name server addresses are specified.

Command Mode

Global configuration

Example

The following example specifies host 131.108.1.111 as the primary name server and host 131.108.1.2 as the secondary server:

```
ip name-server 131.108.1.111 131.108.1.2
```

This command will be reflected in the configuration file as follows:

```
ip name-server 131.108.1.111  
ip name-server 131.108.1.2
```

Related Commands

ip domain-lookup

ip domain-name

ip netmask-format

To specify the format in which network masks are displayed in **show** command output, use the **ip netmask-format** line configuration command. To restore the default display format, use the **no** form of this command.

```
ip netmask-format { bitcount | decimal | hexadecimal }  
no ip netmask-format [bitcount | decimal | hexadecimal]
```

Syntax Description

bitcount	Addresses are followed by a slash and the total number of bits in the network mask. For example, 131.108.11.0/24 indicates that the network mask is 24 bits.
decimal	Network masks are displayed in dotted decimal notation (for example, 255.255.255.0).
hexadecimal	Network masks are displayed in hexadecimal format, as indicated by the leading 0X (for example, 0FFFFFFF00).

Default

Network masks are displayed in dotted decimal format.

Command Mode

Line configuration

Usage Guidelines

IP uses a 32-bit mask that indicates which address bits belong to the network and subnetwork fields and which bits belong to the host field. This is called a network mask. By default, **show** commands display an IP address and then its network mask in dotted decimal notation. For example, a subnet would be displayed as 131.108.11.0 255.255.255.0.

However, you can specify that the display of the network mask appear in hexadecimal format or bit count format instead. The hexadecimal format is commonly used on UNIX systems. The preceding example would be displayed as 131.108.11.0 0FFFFFFF00.

The bitcount format for displaying network masks is to append a slash (/) and the total number of bits in the network mask to the address itself. The above example would be displayed as 131.108.11.0/24.

Example

The following example configures network masks for the specified line to be displayed in bitcount notation in the output of **show** commands:

```
line vty 0 4  
ip netmask-format bitcount
```

ip nhrp authentication

To configure the authentication string for an interface using Next Hop Resolution Protocol (NHRP), use the **ip nhrp authentication** interface configuration command. To remove the authentication string, use the **no** form of this command.

```
ip nhrp authentication string  
no ip nhrp authentication [string]
```

Syntax Description

string Authentication string configured for the source and destination stations that controls whether NHRP stations allow intercommunication. The string can be up to 8 characters long.

Default

No authentication string is configured; the communication server adds no authentication option to NHRP packets it generates.

Command Mode

Interface configuration

Usage Guidelines

All communication servers configured with NHRP on a fabric (for an interface) must share the same authentication string.

Example

In the following example, the authentication string *specialxx* must be configured in all communication servers using NHRP on the interface before NHRP communication occurs:

```
ip nhrp authentication specialxx
```

ip nhrp holdtime

To change the number of seconds that NHRP nonbroadcast, multiaccess (NBMA) addresses are advertised as valid in authoritative NHRP responses, use the **ip nhrp holdtime** interface configuration command. To restore the default value, use the **no** form of this command.

```
ip nhrp holdtime seconds-positive [seconds-negative]  
no ip nhrp holdtime [seconds-positive [seconds-negative]]
```

Syntax Description

<i>seconds-positive</i>	Time in seconds that NBMA addresses are advertised as valid in positive authoritative NHRP responses.
<i>seconds-negative</i>	(Optional) Time in seconds that NBMA addresses are advertised as valid in negative authoritative NHRP responses.

Default

7200 seconds (2 hours) for both arguments

Command Mode

Interface configuration

Usage Guidelines

The **ip nhrp holdtime** command affects authoritative responses only. The advertised holding time is the length of time the communication server tells other communication servers to keep information that it is providing in authoritative NHRP responses. The cached IP-to-NBMA address mapping entries are discarded after the holding time expires.

The NHRP cache can contain static and dynamic entries. The static entries never expire. Dynamic entries expire regardless of whether they are authoritative or nonauthoritative.

If you want to change the valid time period for negative NHRP responses, you must also include a value for positive NHRP responses, because the arguments are position dependent.

Examples

In the following example, NHRP NBMA addresses are advertised as valid in positive authoritative NHRP responses for one hour:

```
ip nhrp holdtime 3600
```

In the following example, NHRP NBMA addresses are advertised as valid in negative authoritative NHRP responses for one hour and in positive authoritative NHRP responses for two hours:

```
ip nhrp holdtime 7200 3600
```

ip nhrp interest

To control which IP packets can trigger sending a Next Hop Resolution Protocol (NHRP) request, use the **ip nhrp interest** interface configuration command. To restore the default value, use the **no** form of this command.

```
ip nhrp interest access-list-number  
no ip nhrp interest [access-list-number]
```

Syntax Description

access-list-number Standard or extended IP access list number in the range 1 through 199.

Default

All non-NHRP packets can trigger NHRP requests.

Command Mode

Interface configuration

Usage Guidelines

Use this command with the **access-list** command to control which IP packets trigger NHRP requests.

Example

In the following example, any TCP traffic can cause NHRP Requests to be sent, but no other IP packets will cause NHRP requests.

```
ip nhrp interest 101  
access-list 101 permit tcp any any
```

Related Commands

access-list (standard)
access-list (extended)

ip nhrp map

To statically configure the IP-to-NBMA address mapping of IP destinations connected to a nonbroadcast, multiaccess (NBMA) network, use the **ip nhrp map** interface configuration command. To remove the static entry from NHRP cache, use the **no** form of this command.

```
ip nhrp map ip-address nbma-address  
no ip nhrp map ip-address nbma-address
```

Syntax Description

<i>ip-address</i>	IP address of the destinations reachable through the NBMA network. This address is mapped to the NBMA address.
<i>nbma-address</i>	Nonbroadcast, multiaccess (NBMA) address which is directly reachable through the NBMA network. The address format varies depending on the medium you are using. For example, ATM has an NSAP address, Ethernet has a MAC address, and SMDS has an E.164 address. This address is mapped to the IP address.

Default

No static IP-to-NBMA cache entries exist.

Command Mode

Interface configuration

Usage Guidelines

You will probably have to configure at least one static mapping in order to reach the NHRP server. Repeat this command to statically configure multiple IP-to-NBMA address mappings.

Example

In the following example, this station in a multipoint tunnel network is statically configured to be served by two NHRP servers 100.0.0.1 and 100.0.1.3. The NBMA address for 100.0.0.1 is statically configured to be 11.0.0.1 and the NBMA address for 100.0.1.3 is 12.2.7.8.

```
interface tunnel 0  
ip nhrp nhs 100.0.0.1  
ip nhrp nhs 100.0.1.3  
ip nhrp map 100.0.0.1 11.0.0.1  
ip nhrp map 100.0.1.3 12.2.7.8
```

Related Command

clear ip nhrp

ip nhrp map multicast

To configure NBMA addresses used as destinations for broadcast or multicast packets to be sent over a tunnel network, use the **ip nhrp map multicast** interface configuration command. To remove the destinations, use the **no** form of this command.

```
ip nhrp map multicast nbma-address  
no ip nhrp map multicast nbma-address
```

Syntax Description

<i>nbma-address</i>	Nonbroadcast, multiaccess (NBMA) address that is directly reachable through the NBMA network. The address format varies depending on the medium you are using.
---------------------	--

Default

No NBMA addresses are configured as destinations for broadcast or multicast packets.

Command Mode

Interface configuration

Usage Guidelines

This command applies to tunnel interfaces only.

This command is useful for supporting broadcasts over a tunnel network when the underlying network does not support IP multicast. If the underlying network does support IP multicast, you should use the **tunnel destination** command to configure a multicast destination for transmission of tunnel broadcasts or multicasts.

When multiple NBMA addresses are configured, the system replicates the broadcast packet for each address.

Example

In the following example, if a packet is sent to 10.255.255.255, it is replicated to destinations 11.0.0.1 and 11.0.0.2. Addresses 11.0.0.1 and 11.0.0.2 are the IP addresses of two other communication servers that are part of the tunnel network, but those addresses are their addresses in the underlying network, not the tunnel network. They would have tunnel addresses that are in network 10.0.0.0.

```
interface tunnel 0  
ip address 10.0.0.3 255.0.0.0  
ip nhrp map multicast 11.0.0.1  
ip nhrp map multicast 11.0.0.2
```

ip nhrp network-id

To enable the Next Hop Resolution Protocol (NHRP) on an interface, use the **ip nhrp network-id** interface configuration command. To disable NHRP on the interface, use the **no** form of this command.

```
ip nhrp network-id number  
no ip nhrp network-id [number]
```

Syntax Description

number Globally unique, 32-bit network identifier for a nonbroadcast, multiaccess (NBMA) network. The range is 1 to 4,294,967,295.

Default

NHRP is disabled on the interface.

Command Mode

Interface configuration

Usage Guidelines

In general, all NHRP stations within a fabric must be configured with the same network identifier.

Example

In the following example, NHRP is enabled on the interface.

```
ip nhrp network-id 1
```


ip nhrp nhs

To specify the address of one or more NHRP Next Hop servers, use the **ip nhrp nhs** interface configuration command. To remove the address, use the **no** form of this command.

```
ip nhrp nhs nhs-address [net-address [netmask]]  
no ip nhrp nhs nhs-address [net-address [netmask]]
```

Syntax Description

<i>nhs-address</i>	Address of the Next Hop server being specified.
<i>net-address</i>	(Optional) IP address of a network served by the Next Hop server.
<i>netmask</i>	(Optional) IP network mask to be associated with the <i>net</i> IP address. The <i>net</i> IP address is logically ANDed with the mask.

Default

No Next Hop servers are explicitly configured, so NHRP fabric mode is assumed and normal IP routing decisions are used to forward NHRP traffic.

Command Mode

Interface configuration

Usage Guidelines

Use this command to specify the address of a Next Hop server and the networks it serves. When Next Hop servers are configured, server mode is assumed. In server mode, each Next Hop server should be configured with information as to what networks are served by the other Next Hop servers in the nonbroadcast, multiaccess (NBMA) network.

For any Next Hop server that is configured, you can specify the multiple networks that it serves by repeating this command with the same *nhs-address* address, but different *net-address* IP network addresses.

If no Next Hop server is configured for an NBMA network, the NHRP fabric mode is assumed.

Example

In the following example, the Next Hop server with address 131.108.10.11 serves IP network 10.0.0.0. The network mask is 255.0.0.0.

```
ip nhrp nhs 131.108.10.11 10.0.0.0 255.0.0.0
```

ip nhrp record

To reenable the use of forward record and reverse record options in NHRP request and reply packets, use the **ip nhrp record** interface configuration command. To suppress the use of such options, use the **no** form of this command.

ip nhrp record
no ip nhrp record

Syntax Description

This command has no arguments or keywords.

Default

Forward record and reverse record options are used in NHRP request and reply packets.

Command Mode

Interface configuration

Usage Guidelines

Forward record and reverse record options provide loop detection and are enabled by default. Using the **no** form of this command disables this method of loop detection. For another method of loop detection, see the **ip nhrp responder** command.

Example

In the following example, forward record and reverse record options are suppressed:

```
no ip nhrp record
```

Related Command

ip nhrp responder

ip nhrp responder

To designate which interface's primary IP address the Next Hop server will use in NHRP Reply packets when the NHRP requestor uses the Responder Address option, use the **ip nhrp responder** interface configuration command. To remove the designation, use the **no** form of this command.

```
ip nhrp responder type number  
no ip nhrp responder [type] [number]
```

Syntax Description

<i>type</i>	Interface type whose primary IP address is used when a Next Hop server complies with a Responder Address option (for example, serial , tunnel).
<i>number</i>	Interface number whose primary IP address is used when a Next Hop server complies with a Responder Address option.

Default

The Next Hop server uses the IP address of the interface where the NHRP Request was received.

Command Mode

Interface configuration

Usage Guidelines

If an NHRP requestor wants to know which Next Hop server generates an NHRP Reply packet, it can request that information through the Responder Address option. The Next Hop server that generates the NHRP Reply packet then complies by inserting its own IP address in the Responder Address option of the NHRP Reply. The Next Hop server uses the primary IP address of the specified interface.

If an NHRP Reply packet being forwarded by a Next Hop server contains that Next Hop server's own IP address, the Next Hop server generates an Error Indication of type "NHRP Loop Detected" and discards the Reply.

Example

In the following example, any NHRP requests for the Responder Address will cause this communication server acting as a Next Hop server to supply the primary IP address of serial interface 0 in the NHRP Reply packet:

```
ip nhrp responder serial 0
```

ip probe proxy

To enable the HP Probe Proxy support, which allows a communication server to respond to HP Probe Proxy Name requests, use the **ip probe proxy** interface configuration command. To disable HP Probe Proxy, use the **no** form of this command.

ip probe proxy
no ip probe proxy

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

HP Probe Proxy Name requests are typically used at sites that have HP equipment and are already using HP Probe.

To use the HP Proxy service, you must first enter the host name of the HP host into the host table using the **ip hp-host** global configuration command.

Example

The following example specifies an HP host's name and address, and then enables Probe Proxy:

```
ip hp-host BCWjo 131.108.1.27
interface ethernet 0
ip probe proxy
```

Related Command

ip hp-host

ip proxy-arp

To enable proxy ARP on an interface, use the **ip proxy-arp** interface configuration command. To disable proxy ARP on the interface, use the **no** form of this command.

ip proxy-arp
no ip proxy-arp

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Example

The following example enables proxy ARP on Ethernet interface 0:

```
interface ethernet 0
ip proxy-arp
```

ip redirects

To enable the sending of redirect messages if the router is forced to resend a packet through the same interface on which it was received, use the **ip redirects** interface configuration command. To disable the sending of redirect messages, use the **no** form of this command.

ip redirects
no ip redirects

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Example

The following example enables the sending of IP redirects on Ethernet interface 0:

```
interface ethernet 0  
ip redirects
```

Related Command

show ip redirects

ip route-cache

To control the use of a high-speed switching cache for IP routing as well as the use of autonomous switching, use the **ip route-cache** interface configuration command. To disable fast switching and autonomous switching, use the **no** form of this command.

ip route-cache
no ip route-cache

Syntax Description

This command has no arguments or keywords.

Default

Fast switching is enabled.

Command Mode

Interface configuration

Usage Guidelines

Using the route cache is often called *fast switching*. The route cache allows outgoing packets to be load-balanced on a *per-destination* basis.

Our communication servers generally offer better packet transfer performance when fast switching is enabled, with one exception. On networks using slow serial links (64K and below), disabling fast switching to enable the per-packet load sharing is usually the best choice.

Examples

The following example disables fast switching:

```
no ip route-cache
```

The following example returns the system to its defaults (fast switching enabled):

```
ip route-cache
```

Related Commands

ip cache-invalidate-delay
show ip cache

ip routing

To enable IP routing on the communication server, use the **ip routing** global configuration command. To disable IP routing on the communication server, use the **no** form of this command.

ip routing
no ip routing

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

If the system is running bridging software, the **no ip routing** command turns off IP routing when setting up a system to bridge (as opposed to route) IP packets.

Example

The following example shows how to enable IP routing:

```
ip routing
```


ip security add

To add a basic security option to all outgoing packets, use the **ip security add** interface configuration command. To disable the adding of a basic security option to all outgoing packets, use the **no** form of this command.

ip security add
no ip security add

Syntax Description

This command has no arguments or keywords.

Default

Disabled, when the security level of the interface is “Unclassified Genser” (or unconfigured). Otherwise, the default is enabled.

Command Mode

Interface configuration

Usage Guidelines

If an outgoing packet does not have a security option present, this interface configuration command will add one as the first IP option. The security label added to the option field is the label that was computed for this packet when it first entered the communication server. Because this action is performed after all the security tests have been passed, this label will either be the same as or will fall within the range of the interface.

Example

The following example adds a basic security option to each packet leaving Ethernet interface 0:

```
interface ethernet 0
ip security add
```

Related Commands

ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security aeso

To attach Auxiliary Extended Security Options (AESOs) to an interface, use the **ip security aeso** command. To disable AESO on an interface, use the **no** form of this command.

ip security aeso *source compartment-bits*
no ip security aeso *source compartment-bits*

Syntax Description

<i>source</i>	Extended Security Option (ESO) source. This can be an integer from 0 through 255.
<i>compartment-bits</i>	Compartment bits in hexadecimal.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Compartment bits are specified only if this AESO is to be inserted in a packet. On every incoming packet at this level on this interface, these AESOs should be present.

Beyond being recognized, no further processing of AESO information is performed. AESO contents are not checked and are assumed to be valid if the source is listed in the configurable AESO table.

Configuring any per-interface extended IP Security Option (IPSO) information automatically enables **ip security extended-allowed** (disabled by default).

Example

In the following example, the extended security option source is defined as 5 and the compartments bits are set to 5.

```
interface ethernet 0
 ip security aeso 5 5
```

Related Commands

ip security eso-info
ip security eso-max
ip security eso-min
ip security extended-allowed

ip security dedicated

To set the level of classification and authority on the interface, use the **ip security dedicated** interface configuration command. To reset the interface to the default classification and authorities, use the **no** form of this command.

```
ip security dedicated level authority [authority...]
no ip security dedicated level authority [authority...]
```

Syntax Description

<i>level</i>	Degree of sensitivity of information. The level keywords are listed in Table 18-1.
<i>authority</i>	Organization that defines the set of security levels that will be used in a network. The authority keywords are listed in Table 18-2.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

All traffic entering the system on this interface must have a security option that exactly matches this label. Any traffic leaving via this interface will have this label attached to it.

The following definitions apply to the descriptions of the IP Security Options (IPSO) in this section:

- **level**—The degree of sensitivity of information. For example, data marked TOPSECRET is more sensitive than data marked SECRET. The level keywords and their corresponding bit patterns are shown in Table 18-1.

Table 18-1 IPSO Level Keywords and Bit Patterns

Level Keyword	Bit Pattern
Reserved4	0000 0001
TopSecret	0011 1101
Secret	0101 1010
Confidential	1001 0110
Reserved3	0110 0110
Reserved2	1100 1100
Unclassified	1010 1011
Reserved1	1111 0001

- **authority**—An organization that defines the set of security levels that will be used in a network. For example, the Genser authority consists of level names defined by the U.S. Defense Communications Agency (DCA). The authority keywords and their corresponding bit patterns are shown in Table 18-2.

Table 18-2 IPSO Authority Keywords and Bit Patterns

Authority Keyword	Bit Pattern
Genser	1000 0000
Siop-Esi	0100 0000
DIA	0010 0000
NSA	0001 0000
DOE	0000 1000

- **label**—A combination of a security level and an authority or authorities.

Example

The following example sets a confidential level with Genser authority:

```
ip security dedicated confidential Genser
```

Related Commands

- ip security add**
- ip security extended-allowed**
- ip security first**
- ip security ignore-authorities**
- ip security implicit-labelling**
- ip security multilevel**
- ip security reserved-allowed**
- ip security strip**

ip security eso-info

To configure system-wide defaults for extended IP Security Option (IPSO) information, use the **ip security eso-info** global configuration command. To return to the default settings, use the **no** form of this command.

```
ip security eso-info source compartment-size default-bit
no ip security eso-info source compartment-size default-bit
```

Syntax Description

<i>source</i>	Hexadecimal or decimal value representing the extended IPSO source. This is an integer from 0 through 255.
<i>compartment-size</i>	Maximum number of bytes of compartment information allowed for a particular extended IPSO source. This is an integer from 1 through 16.
<i>default-bit</i>	Default bit value for any unspent compartment bits.

Default

Disabled

Command mode

Global configuration

Usage Guidelines

This command configures Extended Security Option (ESO) information, including Auxiliary Extended Security Option (AESO). Transmitted compartment info is padded to the size specified by the *compartment-size* argument.

Example

In the following example, system-wide defaults for source, compartment size, and the default bit value are set:

```
ip security eso-info 100 5 1
```

Related Commands

ip security eso-max

ip security eso-min

ip security eso-max

To specify the maximum sensitivity level for an interface, use the **ip security eso-max** interface configuration command. To return to the default, use the **no** form of this command.

```
ip security eso-max source compartment-bits  
no ip security eso-max source compartment-bits
```

Syntax Description

<i>source</i>	Extended Security Option (ESO) source. An integer from 1 through 255.
<i>compartment-bits</i>	Compartment bits in hex.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is used to specify the minimum sensitivity level for a particular interface. Before the per interface compartment information for a particular Network Level Extended Security Option (NLESO) source can be configured, the **ip security eso-info** global configuration command must be used to specify the default information.

On every incoming packet on the interface, these extended security options should be resent at the minimum level and should match the configured compartment bits. Every outgoing packet must have these ESOs.

On every packet transmitted or received on this interface, any NLESO sources present in the IP header should be bounded by the minimum sensitivity level and bounded by the maximum sensitivity level configured for the interface.

When transmitting locally generated traffic out this interface, or adding security information (with the **ip security add** command), the maximum compartment bit information can be used to construct the NLESO sources placed in the IP header.

A maximum of 16 NLESO sources can be configured per interface. Due to IP header length restrictions, a maximum of 9 of these NLESO sources appear in the IP header of a packet.

Example

In the following example, the specified ESO source is 240 and the compartment bits are specified as 500:

```
interface ethernet 0  
ip security eso-max 240 500
```

Related Commands

ip security eso-info

ip security eso-min

ip security eso-min

To configure the minimum sensitivity for an interface, use the **ip security eso-min** interface configuration command. To return to the default, use the **no** form of this command.

```
ip security eso-min source compartment-bits  
no ip security eso-min source compartment-bits
```

Syntax Description

<i>source</i>	Extended Security Option (ESO) source. This is an integer from 1 through 255.
<i>compartment-bits</i>	Compartment bits in hexadecimal.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is used to specify the minimum sensitivity level for a particular interface. Before the per-interface compartment information for a particular Network Level Extended Security Option (NLESO) source can be configured, the **ip security eso-info** global configuration command must be used to specify the default information.

On every incoming packet on this interface, these extended security options should be resent at the minimum level and should match the configured compartment bits. Every outgoing packet must have these ESOs.

On every packet transmitted or received on this interface, any NLESO sources present in the IP header should be bounded by the minimum sensitivity level and bounded by the maximum sensitivity level configured for the interface.

When transmitting locally generated traffic out this interface, or adding security information (with the **ip security add** command), the maximum compartment bit information can be used to construct the NLESO sources placed in the IP header.

A maximum of 16 NLESO sources can be configured per interface. Due to IP header length restrictions, a maximum of 9 of these NLESO sources appear in the IP header of a packet.

Example

In the following example, the specified ESO source is 5 and the compartment bits are specified as 5:

```
interface ethernet 0  
ip security eso-min 5 5
```


Related Commands

ip security eso-info

ip security eso-max

ip security extended-allowed

To accept packets on an interface that has an extended security option present, use the **ip security extended-allowed** interface configuration command. To restore the default, use the **no** form of this command.

ip security extended-allowed
no ip security extended-allowed

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Packets containing extended security options are rejected.

Example

The following example allows Ethernet interface 0 to accept packets that have an extended security option present:

```
interface ethernet 0
 ip security extended-allowed
```

Related Commands

ip security add
ip security dedicated
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security first

To prioritize the presence of security options on a packet, use the **ip security first** interface configuration command. To disable this function, use the **no** form of this command.

ip security first
no ip security first

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

If a basic security option is present on an outgoing packet, but it is not the first IP option, then the packet is moved to the front of the options field when this interface configuration command is used.

Example

The following example ensures that, if a basic security option is present in the options field of a packet exiting Ethernet interface 0, the packet is moved to the front of the options field:

```
interface ethernet 0
ip security first
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security ignore-authorities

To have the router ignore the authorities field of all incoming packets, use the **ip security ignore-authorities** interface configuration command. To disable this function, use the **no** form of this command.

ip security ignore-authorities
no ip security ignore-authorities

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

When the packet's authority field is ignored, the value used in place of this field is the authority value declared for the specified interface. IP security ignore-authorities can only be configured on interfaces with dedicated security levels.

Example

The following example causes Ethernet interface 0 to ignore the authorities field on all incoming packets:

```
interface ethernet 0
ip security ignore-authorities
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security implicit-labelling

To force the router to accept packets on the interface, even if they do not include a security option, use the **ip security implicit-labelling** interface configuration command. To disable this function, use the **no** form of this command.

```
ip security implicit-labelling [level authority [authority...]]  
no ip security implicit-labelling [level authority [authority...]]
```

Syntax Description

<i>level</i>	(Optional) Degree of sensitivity of information. If your interface has multilevel security set, you must specify this argument. The level keywords are listed in Table 18-1 (see the ip security dedicated interface configuration command).
<i>authority</i>	(Optional) Organization that defines the set of security levels that will be used in a network. If your interface has multilevel security set, you must specify this argument. You can specify more than one. The authority keywords are listed in Table 18-2 (see the ip security dedicated interface configuration command).

Default

Enabled, when the security level of the interface is “Unclassified Genser” (or unconfigured). Otherwise, the default is disabled.

Command Mode

Interface configuration

Usage Guidelines

If your interface has multilevel security set, you must use the expanded form of the command (with the optional arguments as noted in brackets) because the arguments are used to specify the precise level and authority to use when labeling the packet. If your interface has dedicated security set, the additional arguments are ignored.

Example

In the following example, an interface is set for security and will accept unlabeled packets:

```
ip security dedicated confidential genser  
ip security implicit-labelling
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities

ip security multilevel
ip security reserved-allowed
ip security strip

ip security multilevel

To set the range of classifications and authorities on an interface, use the **ip security multilevel** interface configuration command. To disable this function, use the **no** form of this command.

```
ip security multilevel level1 [authority1...] to level2 authority2 [authority2...]
no ip security multilevel
```

Syntax Description

<i>level1</i>	Degree of sensitivity of information. The classification level of incoming packets must be equal to or greater than this value for processing to occur. The level keywords are found in Table 18-1 (see the ip security dedicated command).
<i>authority1</i>	(Optional) Organization that defines the set of security levels that will be used in a network. The authority bits must be a superset of this value. The authority keywords are listed in Table 18-2 (see the ip security dedicated command).
to	Separates the range of classifications and authorities.
<i>level2</i>	Degree of sensitivity of information. The classification level of incoming packets must be equal to or less than this value for processing to occur. The level keywords are found in Table 18-1 (see the ip security dedicated command).
<i>authority2</i>	Organization that defines the set of security levels that will be used in a network. The authority bits must be a proper subset of this value. The authority keywords are listed in Table 18-2 (see the ip security dedicated command).

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

All traffic entering or leaving the system must have a security option that falls within this range. Being within range requires that the following two conditions be met:

- The classification level must be greater than or equal to *level1* and less than or equal to *level2*.
- The authority bits must be a superset of *authority1* and a proper subset of *authority2*. That is, *authority1* specifies those authority bits that are required on a packet, while *authority2* specifies the required bits plus any optional authorities that also can be included. If the *authority1* field is the empty set, then a packet is required to specify any one or more of the authority bits in *authority2*.

Example

The following example specifies levels Unclassified to Secret and NSA authority:

```
ip security multilevel unclassified to secret nsa
```

Related Commands

ip security add

ip security dedicated

ip security extended-allowed

ip security first

ip security ignore-authorities

ip security implicit-labelling

ip security reserved-allowed

ip security strip

ip security reserved-allowed

To treat as valid any packets that have Reserved1 through Reserved4 security levels, use the **ip security reserved-allowed** interface configuration command. To disable this feature, use the **no** form of this command.

```
ip security reserved-allowed  
no ip security reserved-allowed
```

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

When you set multilevel security on an interface, and indicate, for example, that the highest range allowed is Confidential, and the lowest is Unclassified, the communication server neither allows nor operates on packets that have security levels of Reserved3 and Reserved2 because they are undefined.

If you use the IP Security Option (IPSO) to block transmission out of unclassified interfaces, and you use one of the Reserved security levels, you *must* enable this feature to preserve network security.

Example

The following example allows a security level of Reserved through Ethernet interface 0:

```
interface ethernet 0  
ip security reserved-allowed
```

Related Commands

```
ip security add  
ip security dedicated  
ip security extended-allowed  
ip security first  
ip security ignore-authorities  
ip security implicit-labelling  
ip security multilevel  
ip security strip
```

ip security strip

To remove any basic security option on outgoing packets on an interface, use the **ip security strip** interface configuration command. To disable this function, use the **no** form of this command.

ip security strip
no ip security strip

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This procedure is performed after all security tests in the communication server have been passed. This command is not allowed for multilevel interfaces.

Example

The following example removes any basic security options on outgoing packets on Ethernet interface 0:

```
interface ethernet 0
ip security strip
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed

ip source-route

To allow the router to handle IP datagrams with source routing header options, use the **ip source-route** global configuration command. To have the router discard any IP datagram containing a source-route option, use the **no** form of this command.

ip source-route
no ip source-route

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Example

The following example enables the handling of IP datagrams with source routing header options:

```
ip source-route
```

Related Command

ping

ip subnet-zero

To enable the use of subnet zero for interface addresses and routing updates, use the **ip subnet-zero** global configuration command. To restore the default, use the **no** form of this command.

ip subnet-zero
no ip subnet-zero

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **ip subnet-zero** command provides the ability to configure and route to subnet-zero subnets.

Subnetting with a subnet address of zero is discouraged because of the confusion inherent in having a network and a subnet with indistinguishable addresses.

Example

In the following example, subnet-zero is enabled for the communication server:

```
ip subnet-zero
```

ip tcp compression-connections

To specify the total number of header compression connections that can exist on an interface, use the **ip tcp compression-connections** interface configuration command. To restore the default, use the **no** form of this command.

```
ip tcp compression-connections number  
no ip tcp compression-connections number
```

Syntax Description

number Number of connections the cache supports. It can be a number from 3 and 256, inclusive.

Default

16 connections

Command Mode

Interface configuration

Usage Guidelines

You should configure one connection for each TCP connection through the specified interface.

Each connection sets up a compression cache entry, so you are in effect specifying the maximum number of cache entries and the size of the cache. Too few cache entries for the specified interface can lead to degraded performance, while too many cache entries can lead to wasted memory.

Note Both ends of the serial connection must use the same number of cache entries.

Example

In the following example, the first serial interface is set for header compression with a maximum of ten cache entries:

```
interface serial 0  
ip tcp header-compression  
ip tcp compression-connections 10
```

Related Commands

```
ip tcp header-compression  
show ip tcp header-compression
```

ip tcp header-compression

To enable TCP header compression, use the **ip tcp header-compression** interface configuration command. To disable compression, use the **no** form of this command.

```
ip tcp header-compression /passive/  
no ip tcp header-compression /passive/
```

Syntax Description

passive (Optional) Compresses outgoing TCP packets only if incoming TCP packets on the same interface are compressed. If you do not specify the **passive** keyword, the communication server compresses all traffic.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

You can compress the headers of your TCP/IP packets in order to reduce the size of your packets. TCP header compression is supported on serial lines using HDLC or PPP encapsulation. You must enable compression on both ends of a serial connection. RFC 1144 specifies the compression process. Compressing the TCP header can speed up Telnet connections dramatically. In general, TCP header compression is advantageous when your traffic consists of many small packets, not for traffic that consists of large packets. Transaction processing (usually using terminals) tends to use small packets while file transfers use large packets. This feature only compresses the TCP header, so it has no effect on UDP packets or other protocol headers.

When compression is enabled, fast switching is disabled. This means that fast interfaces like T1 can overload the router. Consider your network's traffic characteristics before using this command.

Example

In the following example, the first serial interface is set for header compression with a maximum of ten cache entries:

```
interface serial 0  
ip tcp header-compression  
ip tcp compression-connections 10
```

Related Commands

```
ip tcp compression-connections  
show ip tcp header-compression
```

ip tcp synwait-time

To set a period of time the router waits while attempting to establish a TCP connection before it times out, use the **ip tcp synwait-time** global configuration command. To restore the default time, use the **no** form of this command.

```
ip tcp synwait-time seconds  
no ip tcp synwait-time seconds
```

Syntax Description

seconds Time in seconds the communication server waits while attempting to establish a TCP connection. It can be an integer from 5 to 300 seconds. The default is 30 seconds.

Default

30 seconds

Command Mode

Global configuration

Usage Guidelines

In previous versions of communication server software, the system would wait a fixed 30 seconds when attempting to establish a TCP connection. If your network contains Public Switched Telephone Network dial-on-demand routing (PSTN DDR), it is possible that the call setup time will exceed 30 seconds. This amount of time is not sufficient in networks that have dial-up asynchronous connections because it will affect your ability to Telnet over the link (from the communication server) if the link must be brought up. If you have this type of network, you might want to set this value to the UNIX value of 75.

Because this is a host parameter, it does not pertain to traffic going *through* the communication server, just for traffic originated *at* the communication server. Because UNIX has a fixed 75-second timeout, hosts are unlikely to see this problem.

Example

The following example configures the communication server to continue attempting to establish a TCP connection for 180 seconds:

```
ip tcp synwait-time 180
```

ip unnumbered

To enable IP processing on a serial interface without assigning an explicit IP address to the interface, use the **ip unnumbered** interface configuration command. To disable the IP processing on the interface, use the **no** form of this command.

ip unnumbered *type number*
no ip unnumbered *type number*

Syntax Description

<i>type</i>	Interface type
<i>number</i>	Interface number

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Whenever the unnumbered interface generates a packet (for example, for a routing update), it uses the address of the specified interface as the source address of the IP packet. It also uses the address of the specified interface in determining which routing processes are sending updates over the unnumbered interface. Restrictions include the following:

- Serial interfaces using HDLC, PPP, LAPB, and Frame Relay encapsulations, as well as SLIP and tunnel interfaces can be unnumbered. It is not possible to use this interface configuration command with X.25 or SMDS interfaces.
- You cannot use the **ping EXEC** command to determine whether the interface is up, because the interface has no address. Simple Network Management Protocol (SNMP) can be used to remotely monitor interface status.
- You cannot netboot a runnable image over an unnumbered serial interface.
- You cannot support IP security options on an unnumbered interface.

The interface you specify by the *type* and *number* arguments must be enabled (listed as “up” in the **show interfaces** command display).

If you are configuring IS-IS across a serial line, you should configure the serial interfaces as unnumbered. This allows you to conform with RFC 1195, which states that IP addresses are not required on each interface.

Note Using an unnumbered serial line between different major networks (majornets) requires special care. If at each end of the link there are different majornets assigned to the interfaces you specified as unnumbered, then any routing protocol running across the serial line must not advertise subnet information.

Example

In the following example, the first serial interface is given Ethernet 0's address:

```
interface ethernet 0
ip address 131.108.6.6 255.255.255.0
interface serial 0
ip unnumbered ethernet 0
```

ip unreachable

To enable the generation of ICMP Unreachable messages, use the **ip unreachable** interface configuration command. To disable this function, use the **no** form of this command.

ip unreachable
no ip unreachable

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

If the communication server receives a nonbroadcast packet destined for itself that uses a protocol it does not recognize, it sends an ICMP *Protocol Unreachable* message to the source.

If the communication server receives a datagram that it cannot deliver to its ultimate destination because it knows of no route to the destination address, it replies to the originator of that datagram with an ICMP *Host Unreachable* message.

This command affects all kinds of ICMP unreachable messages.

Example

The following example enables the generation of ICMP Unreachable messages, as appropriate, on an interface:

```
interface ethernet 0
ip unreachable
```

ping (user)

To check host reachability and network connectivity, use the **ping** (IP packet internet groper function) user EXEC command.

```
ping [protocol] {host | address}
```

Syntax Description

<i>protocol</i>	(Optional) Protocol keyword. The default is IP.
<i>host</i>	Host name of system to ping.
<i>address</i>	IP address of system to ping.

Command Mode

EXEC

Usage Guidelines

The **ping** command sends ICMP *Echo* messages. If the communication server receives an ICMP *Echo* message, it sends an ICMP *Echo Reply* message to the source of the ICMP *Echo* message.

The user ping feature provides a basic ping facility for IP users who do not have system privileges. This feature allows the communication server to perform the simple default ping functionality for the IP protocol. Only the nonverbose form of the **ping** command is supported for user pings.

If the system cannot map an address for a host name, it will return an “%Unrecognized host or address” error message.

To abort a ping session, type the escape sequence (by default, Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key).

Table 18-3 describes the test characters that the ping facility sends.

Table 18-3 Ping Test Characters

Char	Description
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	Destination unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
M	Could not fragment.
?	Unknown packet type.

Sample Display Using an IP Host Name

The following display shows sample ping output when you ping a host named fred:

```
cs> ping fred
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.31.7.27, timeout is 2 seconds:
!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/3/4 ms
```

Sample Display Using the Broadcast Address

The following display shows sample ping output when you ping the broadcast address of 255.255.255.255:

```
cs> ping 255.255.255.255
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:

Reply to request 0 from 160.89.48.15 (4 ms)
Reply to request 0 from 160.89.48.10 (4 ms)
Reply to request 0 from 160.89.48.19 (4 ms)
Reply to request 0 from 160.89.49.15 (4 ms)
Reply to request 1 from 160.89.48.15 (4 ms)
Reply to request 1 from 160.89.48.10 (4 ms)
Reply to request 1 from 160.89.48.19 (4 ms)
Reply to request 1 from 160.89.49.15 (4 ms)
Reply to request 2 from 160.89.48.15 (4 ms)
Reply to request 2 from 160.89.48.10 (4 ms)
Reply to request 2 from 160.89.48.19 (4 ms)
Reply to request 2 from 160.89.49.15 (4 ms)
Reply to request 3 from 160.89.48.15 (4 ms)
Reply to request 3 from 160.89.48.10 (4 ms)
Reply to request 3 from 160.89.48.19 (4 ms)
Reply to request 3 from 160.89.49.15 (4 ms)
Reply to request 4 from 160.89.48.15 (4 ms)
Reply to request 4 from 160.89.48.10 (4 ms)
Reply to request 4 from 160.89.48.19 (4 ms)
Reply to request 4 from 160.89.49.15 (4 ms)
```

Related Command

ping (privileged)

ping (privileged)

To check host reachability and network connectivity, use the **ping** (IP packet internet groper function) user EXEC command.

```
ping [protocol] {host | address}
```

Syntax Description

<i>protocol</i>	(Optional) Protocol keyword. The default is IP.
<i>host</i>	Host name of system to ping.
<i>address</i>	IP address of system to ping.

Command Mode

Privileged EXEC

Usage Guidelines

The **ping** command sends ICMP *Echo* messages. If the router receives an ICMP *Echo* message, it sends an ICMP *Echo Reply* message to the source of the ICMP *Echo* message.

You can use the IP **ping** command to diagnose serial line problems. By placing the local or remote CSU/DSU into loopback mode and pinging your own interface, you can isolate the problem to the communication server or leased line.

Multicast and broadcast pings are fully supported. When you ping the broadcast address of 255.255.255.255, the system will send out pings and print a list of all stations responding. You can also ping a local network to get a list of all systems that respond, as in the following example, where 128.111.3 is a local network:

```
ping 128.111.3.255
```

As a side-effect, you also can get a list of all multicast-capable hosts that are connected directly to the communication server from which you are pinging, as in the following example:

```
ping 224.0.0.1
```

To abort a ping session, type the escape sequence (by default, Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key).

Table 18-4 describes the test characters that the ping facility sends.

Table 18-4 Ping Test Characters

Char	Description
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	Destination unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.

Char	Description
M	Could not fragment.
?	Unknown packet type.

You can use the extended command mode of the **ping** command to specify the supported Internet header options, as shown in the following sample display.

Sample Display Showing Extended Command Sequence

To enter **ping** extended command mode, enter **yes** at the extended commands prompt of the **ping** command. The following display shows a sample **ping** extended command sequence.

```
cs# ping
Protocol [ip]:
Target IP address: 192.31.7.27
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address: 131.108.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.31.7.27, timeout is 2 seconds:
!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/3/4 ms
```

Table 18-5 describes significant fields shown in the display.

Table 18-5 IP Ping Internet Header Options Field Descriptions

Field	Description
Protocol [ip]:	Default is IP.
Target IP address:	Prompts for the IP address or host name of the destination node you plan to ping.
Repeat count [5]:	Number of ping packets that will be sent to the destination address. Default: 5.
Datagram size [100]:	Size of the ping packet (in bytes). Default: 100 bytes.
Timeout in seconds [2]:	Timeout interval. Default: 2 (seconds).
Extended commands [n]:	Specifies whether or not a series of additional commands appears. Many of the following displays and tables show and describe these commands. Default: no.
Source address:	IP address that appears in the ping packet as the source address.
Type of service [0]:	Internet service quality selection. See RFC 791 for more information. Default: 0.

Field	Description
Set DF bit in IP header?	Don't Fragment. Specifies that if the packet encounters a node in its path that is configured for a smaller MTU than the packet's MTU, that the packet is to be dropped and an error message is to be sent to the communication server at the packet's source address. If performance problems are encountered on the network, a node configured for a small MTU could be a contributing factor. This feature can be used to determine the smallest MTU in the path. Default: no.
Data pattern [0xABCD]:	Sets 16-bit hexadecimal data pattern. Default: 0xABCD. Varying the data pattern in this field (to all ones or all zeros for example) can be useful when debugging data sensitivity problems on CSU/DSUs, or detecting cable-related problems such as cross talk.
Loose, Strict, Record, Timestamp, Verbose [none]:	Supported Internet header options. The communication server examines the header options to every packet that passes through it. If it finds a packet with an invalid option, the communication server sends an ICMP <i>Parameter Problem</i> message to the source of the packet and discards the packet. The Internet header options follow: <ul style="list-style-type: none"> • Loose • Strict • Record—See the following section for more information on this helpful option. • Timestamp • Verbose Default: none. For more information on these header options, see RFC 791.
Sweep range of sizes [n]:	Allows you to vary the sizes of the echo packets being sent. This capability is useful for determining the minimum sizes of the MTUs configured on the nodes along the path to the destination address. Packet fragmentation contributing to performance problems can then be reduced.
!!!!	Each exclamation point (!) indicates receipt of a reply. A period (.) indicates the network server timed out while waiting for a reply. Other characters may appear in the ping output display, depending on the protocol type.
Success rate is 100 percent	Percentage of packets successfully echoed back to the communication server. Anything less than 80 percent is usually considered problematic.
round-trip min/avg/max = 1/3/4 ms	Round-trip travel time intervals for the protocol echo packets, including minimum/average/maximum (in milliseconds).

Use the Record Route Option

Using the Record Route option to trace a path to a particular destination address. Be aware, however, that the **trace EXEC** command performs a similar function, but the latter does not have the nine-hop limitation.

Sample Display Showing the Record Route Option

The following display shows sample extended **ping** output when this option is specified:

```
cs# ping
Protocol [ip]:
Target IP address: fred
```

```
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address:
Type of service [0]:
Set DF bit in IP header? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: r
Number of hops [ 9 ]:
Loose, Strict, Record, Timestamp, Verbose[RV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 131.108.1.115, timeout is 2 seconds:
Packet has IP options: Total option bytes= 39, padded length=40
Record route: <*> 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
                0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
```

The following display is a detail of the Echo packet section:

```
0 in 4 ms. Received packet has options
Total option bytes= 40, padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
                131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
End of list

1 in 8 ms. Received packet has options
Total option bytes= 40, padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.6 131.108.1.115
                131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
End of list

2 in 4 ms. Received packet has options
Total option bytes= 40, padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
End of list

3 in 8 ms. Received packet has options
Total option bytes= 40, padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.6 131.108.1.115
                131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
End of list

4 in 4 ms. Received packet has options
Total option bytes= 40, padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
                131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
End of list

Success rate is 100 percent, round-trip min/avg/max = 4/5/8 ms
cs#
```

In this display, five ping echo packets are sent to the destination address 131.108.1.115. The echo packet detail section includes specific information about each of these echo packets.

The lines of **ping** output that are unique when the Record Route option is specified are described as follows.

The following line of output allows you to specify the number of hops that will be recorded in the route. Range: 1 through 9. Default: 9.

```
Number of hops [ 9 ]:
```


The following line of output indicates that IP header options have been enabled on the outgoing echo packets and shows the number of option bytes and padded bytes in the headers of these packets.

```
Packet has IP options: Total option bytes= 39, padded length=40
```

The following lines of output indicate that the fields that will contain the IP addresses of the nodes in the routes have been zeroed out in the outgoing packets.

```
Record route: <*> 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
               0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
```

The following lines of output display statistics for the first of the five echo packets sent. 0 is the number assigned to this packet to indicate that it is the first in the series. 4 ms indicates the round trip travel time for the packet.

```
0 in 4 ms. Received packet has options
Total option bytes= 40, padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
               131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
```

The following line of output indicates that four nodes were included in the packet's route, including the communication server at source address 160.89.80.31, two intermediate nodes at addresses 131.108.6.10 and 131.108.1.7, and the destination node at address 131.108.1.115. The underlined address shows where the original route differs from the return route in the line that follows this line.

```
Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
```

The following line of output includes the addresses of the four nodes in the return path of the echo packet. The underlined address shows where the return route differs from the original route shown in the previous line of output.

```
131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
```

Related Command

ping (user)

show access-lists

To display the contents of all current access lists, use the **show access-lists** privileged EXEC command.

show access-lists

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show access-lists** command:

```
cs# show access-lists
Standard IP access list 19
  permit 131.108.19.0
  deny 0.0.0.0, wildcard bits 255.255.255.255
Standard IP access list 49
  permit 131.108.31.0, wildcard bits 0.0.0.255
  permit 131.108.194.0, wildcard bits 0.0.0.255
  permit 131.108.195.0, wildcard bits 0.0.0.255
  permit 131.108.196.0, wildcard bits 0.0.0.255
  permit 131.108.197.0, wildcard bits 0.0.0.255
Extended IP access list 101
  permit tcp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255 eq 23
Type code access list 201
  permit 0x6001 0x0000
Type code access list 202
  permit 0x6004 0x0000
  deny 0x0000 0xFFFF
```

For information on how to configure access lists, refer to the chapter “Configuring IP” of the *Access and Communication Servers Configuration Guide*.

Related Command

access-list

show arp

To display the entries in the ARP table for the router, use the **show arp** privileged EXEC command.

show arp

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show arp** command:

```
cs# show arp

Protocol    Address          Age (min)    Hardware Addr  Type   Interface
-----
Internet    131.108.42.112   120         0000.a710.4baf ARPA   Ethernet3
AppleTalk    4028.5           29          0000.0c01.0e56 SNAP   Ethernet2
Internet    131.108.42.114   105         0000.a710.859b ARPA   Ethernet3
AppleTalk    4028.9           -           0000.0c02.a03c SNAP   Ethernet2
Internet    131.108.42.121   42          0000.a710.68cd ARPA   Ethernet3
Internet    131.108.36.9     -           0000.3080.6fd4 SNAP   TokenRing0
AppleTalk    4036.9           -           0000.3080.6fd4 SNAP   TokenRing0
Internet    131.108.33.9     -           0000.0c01.7bbd SNAP   Fddi0
```

Table 18-6 describes significant fields shown in the first line of output in the display.

Table 18-6 Show ARP Field Descriptions

Field	Description
Protocol	Indicates the type of network address this entry includes.
Address	Network address that is mapped to the MAC address in this entry.
Age (min)	Indicates the interval (in minutes) since this entry was entered in the table, rather than the interval since the entry was last used. (The timeout value is 4 hours.)
Hardware Addr	MAC address mapped to the network address in this entry.
Type	Indicates the encapsulation type the communication server is using for the network address in this entry. Possible values include: <ul style="list-style-type: none"> • ARPA • SNAP • ETLK (EtherTalk) • SMDS
Interface	Indicates the interface associated with this network address.

show dnsix

To display state information and the current configuration of the DNSIX audit writing module, use the **show dnsix** privileged EXEC command.

show dnsix

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show dnsix** command:

```
cs1# show dnsix
  Audit Trail Enabled with Source 128.105.2.5
    State: PRIMARY
    Connected to 128.105.2.4
    Primary 128.105.2.4
    Transmit Count 1
    DMDP retries 4
    Authorization Redirection List:
      128.105.2.4
    Record count: 0
    Packet Count: 0
    Redirect Rcv: 0
```

show hosts

To display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of host names and addresses, use the **show hosts EXEC** command.

show hosts

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show hosts** command:

```
cs# show hosts

Default domain is CISCO.COM
Name/address lookup uses domain service
Name servers are 255.255.255.255
Host          Flag          Age   Type          Address(es)
SLAG.CISCO.COM (temp, OK)    1     IP            131.108.4.10
CHAR.CISCO.COM (temp, OK)    8     IP            192.31.7.50
CHAOS.CISCO.COM (temp, OK)    8     IP            131.108.1.115
DIRT.CISCO.COM (temp, EX)    8     IP            131.108.1.111
DUSTBIN.CISCO.COM (temp, EX)    0     IP            131.108.1.27
DREGS.CISCO.COM (temp, EX)    24    IP            131.108.1.30
```

Table 18-7 describes significant fields shown in the display.

Table 18-7 Show Hosts Field Descriptions

Field	Description
Flag	A temporary entry is entered by a name server; the communication server removes the entry after 72 hours of inactivity. A perm entry is entered by a configuration command and is not timed out. Entries marked OK are believed to be valid. Entries marked ?? are considered suspect and subject to revalidation. Entries marked EX are expired.
Age	Indicates the number of hours since the communication server last referred to the cache entry.
Type	Identifies the type of address, for example, IP, CLNS, or X.121. If you have used the ip hp-host global configuration command, the show hosts command will display these host names as type HP-IP.
Address(es)	Shows the address of the host. One host may have up to eight addresses.

Related Command

clear host

show ip access-list

To display the contents of all current IP access lists, use the **show ip access-list** EXEC command.

```
show ip access-list [access-list-number]
```

Syntax Description

access-list-number (Optional) Number of the IP access list to display. This is a decimal number from 1 to 199.

Defaults

Displays all standard and extended IP access lists.

Command Mode

EXEC

Usage Guidelines

The **show ip access-list** command provides output identical to the **show access-lists** command, except that it is IP-specific and allows you to specify a particular access list.

Sample Display

The following is sample output from the **show ip access-list** command:

```
cs# show ip access-list  
Extended IP access list 101  
  deny udp any any eq ntp  
  permit tcp any any  
  permit udp any any eq tftp  
  permit icmp any any  
  permit udp any any eq domain
```

show ip accounting

To display the active accounting or checkpointed database, use the **show ip accounting** privileged EXEC command.

```
show ip accounting [checkpoint] [output-packets / access-violations]
```

Syntax Description

checkpoint	(Optional) Indicates that the checkpointed database should be displayed.
output-packets	(Optional) Indicates that information pertaining to packets that passed access control and were successfully routed should be displayed. This is the default value, if neither output-packets nor access-violations is specified.
access-violations	(Optional) Indicates that information pertaining to packets that failed access lists and were not routed should be displayed.

Command Mode

EXEC

Sample Display

To use this command, you must first enable IP accounting on a per-interface basis.

Following is sample output from the **show ip accounting** command:

```
cs# show ip accounting

      Source           Destination           Packets           Bytes
131.108.19.40         192.67.67.20          7                306
131.108.13.55         192.67.67.20          67               2749
131.108.2.50          192.12.33.51          17               1111
131.108.2.50          130.93.2.1             5                319
131.108.2.50          130.93.1.2             463              30991
131.108.19.40         130.93.2.1             4                262
131.108.19.40         130.93.1.2             28               2552
131.108.20.2          128.18.6.100          39               2184
131.108.13.55         130.93.1.2             35               3020
131.108.19.40         192.12.33.51          1986             95091
131.108.2.50          192.67.67.20          233              14908
131.108.13.28         192.67.67.53          390              24817
131.108.13.55         192.12.33.51          214669           9806659
131.108.13.111        128.18.6.23           27739            1126607
131.108.13.44         192.12.33.51          35412            1523980
192.31.7.21           130.93.1.2             11               824
131.108.13.28         192.12.33.2            21               1762
131.108.2.166         192.31.7.130          797              141054
131.108.3.11          192.67.67.53           4                246
192.31.7.21           192.12.33.51          15696            695635
192.31.7.24           192.67.67.20           21               916
131.108.13.111        128.18.10.1            16               1137
```

Table 18-8 describes significant fields shown in the display.

Table 18-8 Show IP Accounting Field Descriptions

Field	Description
Source	Source address of the packet.
Destination	Destination address of the packet.
Packets	Number of packets transmitted from the source address to the destination address.
Bytes	Number of bytes transmitted from the source address to the destination address.

Following is sample output from the **show ip accounting access-violations** command:

```

cs# show ip accounting

      Source      Destination      Packets      Bytes      ACL 131.108.62.69
131.108.13.55    10              67           2749
131.108.2.50     9               17           1111
131.108.2.50     9               5            319
131.108.2.50     9              463          30991
131.108.19.40   10              4            262
131.108.19.40   10              28           2552
Accounting data age is 41
    
```

Table 18-9 describes significant fields shown in the display.

Table 18-9 Show IP Accounting Access-Violation Field Descriptions

Field	Description
Source	Source address of the packet.
Destination	Destination address of the packet.
Packets	Number of packets transmitted from the source address to the destination address that violated the access control list.
Bytes	Number of bytes transmitted from the source address to the destination address that violated the access-control list.
ACL	Number of the access list that the transmitted packets violated.

Related Commands

- clear ip accounting**
- ip accounting**
- ip accounting-list**
- ip accounting-threshold**
- ip accounting-transits**

show ip aliases

To display the router's Internet addresses mapped to TCP ports (aliases) and SLIP addresses, which are treated similarly to aliases, use the **show ip aliases** EXEC command.

show ip aliases

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

To distinguish a SLIP address from a normal alias address, the command output uses the form SLIP TTY1 for the "port" number, where 1 is the auxiliary port.

Sample Display

The following is sample output from the **show ip aliases** command:

```
cs# show ip aliases

      IP Address      Port
131.108.29.245     SLIP TTY1
```

The display lists the IP address and corresponding port number.

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show line †

show ip arp

To display the Address Resolution Protocol (ARP) cache, where SLIP addresses appear as permanent ARP table entries, use the **show ip arp** EXEC command.

show ip arp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

ARP establishes correspondences between network addresses (an IP address, for example) and LAN hardware addresses (Ethernet addresses). A record of each correspondence is kept in a cache for a predetermined amount of time and then discarded.

Sample Display

The following is sample output from the **show ip arp** command:

```
cs# show ip arp

Protocol Address          Age (min)  Hardware Addr  Type   Interface
-----
Internet 131.108.62.192      187       0800.2010.a3b6  ARPA   Ethernet3
Internet 131.108.62.245      68        0800.200e.28f8  ARPA   Ethernet3
Internet 131.108.1.140       139       0000.0c01.2812  ARPA   Ethernet0
Internet 131.108.62.160     187       0800.200e.4dab  ARPA   Ethernet3
Internet 131.108.1.111      27        0800.2007.8866  ARPA   Ethernet0
Internet 131.108.1.117     119       0000.0c00.f346  ARPA   Ethernet0
Internet 131.108.1.115      28        0000.0c01.0509  ARPA   Ethernet0
Internet 131.108.1.77       1         0800.200e.57ce  ARPA   Ethernet0
Internet 192.31.7.29        225       aa00.0400.0234  ARPA   Ethernet2
Internet 192.31.7.17        118       2424.c01f.0711  ARPA   Ethernet2
Internet 192.31.7.18        135       0000.0c01.2817  ARPA   Ethernet2
Internet 192.31.7.21        119       2424.c01f.0715  ARPA   Ethernet2
Internet 131.108.1.33       1         0800.2008.c52e  ARPA   Ethernet0
Internet 131.108.62.1       -         0000.0c00.750f  ARPA   Ethernet3
Internet 131.108.31.35     119       0800.2010.8c5b  ARPA   Ethernet7
Internet 131.108.62.7       14        0000.0c00.33ce  ARPA   Ethernet3
Internet 131.108.1.55      155       0800.200e.e443  ARPA   Ethernet0
```

Table 18-10 describes significant fields shown in the display.

Table 18-10 Show IP ARP Field Displays

Field	Description
Protocol	Protocol for network address in the Address field.
Address	The network address that corresponds to Hardware Addr.
Age (min)	Age, in minutes, of the cache entry.
Hardware Addr	LAN hardware address a MAC address that corresponds to network address.
Type	Type of encapsulation: <ul style="list-style-type: none">• ARPA—Ethernet• SNAP—RFC 1042• SAP—IEEE 802.3
Interface	Interface to which this address mapping has been assigned.

show ip cache

To display the routing table cache used to fast switch IP traffic, use the **show ip cache EXEC** command.

show ip cache [*prefix mask*] [*interface-type interface-number*]

Syntax Description

<i>prefix</i>	(Optional) Display only the entries in the cache that match the prefix and mask combination.
<i>mask</i>	(Optional) Display only the entries in the cache that match the prefix and mask combination.
<i>interface-type</i>	(Optional) Display only the entries in the cache that match the interface type and number combination.
<i>interface-number</i>	(Optional) Display only the entries in the cache that match the interface type and number combination.

Command Mode

EXEC

Usage Guidelines

The **show ip cache** display shows MAC headers up to 92 bytes.

Sample Displays

The following is sample output from the **show ip cache** command:

```
cs# show ip cache

IP routing cache version 4490, 141 entries, 20772 bytes, 0 hash overflows
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
  quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 0:06:31 ago

Prefix/Length      Age      Interface      MAC Header
131.108.1.1/32     0:01:09  Ethernet0/0    AA000400013400000C0357430800
131.108.1.7/32     0:04:32  Ethernet0/0    00000C01281200000C0357430800
131.108.1.12/32    0:02:53  Ethernet0/0    00000C029FD000000C0357430800
131.108.2.13/32    0:06:22  Fddi2/0        00000C05A3E000000C035753AAAA0300
00000800
131.108.2.160/32   0:06:12  Fddi2/0        00000C05A3E000000C035753AAAA0300
00000800
131.108.3.0/24     0:00:21  Ethernet1/2    00000C026BC600000C03574D0800
131.108.4.0/24     0:02:00  Ethernet1/2    00000C026BC600000C03574D0800
131.108.5.0/24     0:00:00  Ethernet1/2    00000C04520800000C03574D0800
131.108.10.15/32   0:05:17  Ethernet0/2    00000C025FF500000C0357450800
131.108.11.7/32    0:04:08  Ethernet1/2    00000C010E3A00000C03574D0800
131.108.11.12/32   0:05:10  Ethernet0/0    00000C01281200000C0357430800
131.108.11.57/32   0:06:29  Ethernet0/0    00000C01281200000C0357430800
```

Table 18-11 describes significant fields shown in the display.

Table 18-11 Show IP Cache Field Descriptions

Field	Description
IP routing cache version	Version number of this table. This number is incremented any time the table is flushed.
entries	Number of valid entries.
bytes	Number of bytes of processor memory for valid entries.
hash overflows	Number of times autonomous switching cache overflowed.
Minimum invalidation interval	Minimum time delay between cache invalidation request and actual invalidation.
maximum interval	Maximum time delay between cache invalidation request and actual invalidation.
quiet interval	Length of time between cache flush requests before the cache will be flushed.
threshold n requests	Maximum number of requests that can occur while the cache is considered quiet.
Invalidation rate <i>n</i> in last <i>m</i> seconds	Number of cache invalidations during the last <i>m</i> seconds.
0 in last 3 seconds	Number of cache invalidation requests during the last quiet interval.
Last full cache invalidation occurred nn:nn:nn ago	Time since last full cache invalidation was performed.
Prefix/Length	Network reachability information for cache entry.
Age	Age of cache entry.
Interface	Output interface type and number.
MAC Header	Layer 2 encapsulation information for cache entry.

The following is sample output from the **show ip cache** command with a prefix and a network mask specified:

```
cs# show ip cache 131.108.5.0 255.255.255.0

IP routing cache version 4490, 119 entries, 17464 bytes, 0 hash overflows
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
  quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 0:11:56 ago

Prefix/Length      Age      Interface      MAC Header
131.108.5.0/24    0:00:34  Ethernet1/2    0000C0452080000C03574D0800
```

The following is sample output from the **show ip cache** command with an interface specified:

```
cs# show ip cache e0/2

IP routing cache version 4490, 141 entries, 20772 bytes, 0 hash overflows
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
  quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 0:06:31 ago
Prefix/Length      Age      Interface      MAC Header
131.108.10.15/32  0:05:17  Ethernet0/2    0000C025FF50000C0357450800
```

show ip interface

To display the usability status of interfaces, use the **show ip interface** EXEC command.

```
show ip interface [type number]
```

Syntax Description

<i>type</i>	(Optional) Interface type
<i>number</i>	(Optional) Interface number

Command Mode

EXEC

Usage Guidelines

A communication server automatically enters a directly connected route in the routing table if the interface is usable. A usable interface is one through which the communication server can send and receive packets. If the communication server determines that an interface is not usable, it removes the directly connected routing entry from the routing table. Removing the entry allows the communication server to use dynamic routing protocols to determine backup routes to the network (if any).

If the interface can provide two-way communication, the line protocol is marked “up.” If the interface hardware is usable, the interface is marked “up.”

If you specify an optional interface type, you will see only information on that specific interface.

If you specify no optional parameters you will see information on all the interfaces.

When an asynchronous interface is encapsulated with PPP or SLIP, IP fast switching is enabled. A **show ip interface** command on an asynchronous interface encapsulated with PPP or SLIP displays a message indicating that IP fast switching is enabled.

Sample Display

The following is sample output from the **show ip interface** command:

```
cs# show ip interface

Ethernet 0 is up, line protocol is up
  Internet address is 192.54.222.2, subnet mask is 255.255.255.0
  Broadcast address is 192.54.222.0
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is 192.52.71.4
  Secondary address 131.192.115.2, subnet mask 255.255.255.0
  Outgoing access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachable are always sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  Gateway Discovery is disabled
  IP accounting is disabled
```

```
TCP/IP header compression is disabled
Probe proxy name replies are disabled
```

Table 18-12 describes significant fields shown in the display.

Table 18-12 Show IP Interface Field Descriptions

Field	Description
Ethernet 0 is up	If the interface hardware is usable, the interface is marked “up.” For an interface to be usable, both the interface hardware and line protocol must be up.
line protocol is up	If the interface can provide two-way communication, the line protocol is marked “up.” For an interface to be usable, both the interface hardware and line protocol must be up.
Broadcast address	Shows the broadcast address.
Helper address	Specifies a helper address, if one has been set.
Outgoing access list	Indicates whether or not the interface has an outgoing access list set.
Proxy ARP	Indicates whether Proxy ARP is enabled for the interface.
Security level	Specifies the IPSO security level set for this interface.
ICMP redirects	Specifies whether redirects will be sent on this interface.
ICMP unreachable	Specifies whether unreachable messages will be sent on this interface.
ICMP mask replies	Specifies whether mask replies will be sent on this interface.
IP fast switching	Specifies whether fast switching has been enabled for this interface. It is generally enabled on serial interfaces, such as this one.
Gateway Discovery	Specifies whether the discovery process has been enabled for this interface. It is generally disabled on serial interfaces.
IP accounting	Specifies whether IP accounting is enabled for this interface and what the threshold (maximum number of entries) is.
TCP/IP header compression	Indicates whether compression is enabled or disabled.
Probe proxy name	Indicates whether HP Probe proxy name replies are generated.

show ip masks

To display the masks used for network addresses and the number of subnets using each mask, use the **show ip masks** EXEC command.

```
show ip masks address
```

Syntax Description

<i>address</i>	Network address for which a mask is required
----------------	--

Command Mode

EXEC

Usage Guidelines

The **show ip masks** command is useful for debugging when variable-length subnet masks (VLSM) are used. It shows the number of masks associated with the network and the number of routes for each mask.

Sample Display

The following is sample output from the **show ip masks** command:

```
cs# show ip masks 131.108.0.0
Mask           Reference count
255.255.255.255 2
255.255.255.0   3
255.255.0.0     1
```


show ip nhrp

To display the Next Hop Resolution Protocol (NHRP) cache, use the **show ip nhrp** EXEC command.

```
show ip nhrp [dynamic | static] [interface-type interface-number]
```

Syntax Description

dynamic	(Optional) Displays only the dynamic (learned) IP-to-NBMA address cache entries.
static	(Optional) Displays only the static IP-to-NBMA address entries in the cache (configured through the ip nhrp map command).
<i>type</i>	(Optional) Interface type about which to display the NHRP cache (for example, atm , tunnel).
<i>number</i>	(Optional) Interface number about which to display the NHRP cache.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip nhrp** command:

```
cs# show ip nhrp

10.0.0.2 255.255.255.255, ATM0/0 created 0:00:43 expire 1:59:16
  Type: dynamic Flags: authoritative
  NBMA address: 11.1111.1111.1111.1111.1111.1111.1111.1111.1111.11
10.0.0.1 255.255.255.255, Tunnel0 created 0:10:03 expire 1:49:56
  Type: static Flags: authoritative
  NBMA address: 11.1.1.2
cs#
```

Table 18-13 describes the fields in the display.

Table 18-13 Show IP NHRP Field Descriptions

Field	Description
100.0.0.2 255.255.255.255	IP address and its network mask in the IP-to-NBMA address cache. The mask is currently always 255.255.255.255 because we do not support aggregation of NBMA information through NHRP.
ATM0/0 created 0:00:43	Interface type and number (in this case, ATM slot and port numbers) and how long ago it was created (hours:minutes:seconds).
expire 1:59:16	Time in which the positive and negative authoritative NBMA address will expire (hours:minutes:seconds). This value is based on the ip nhrp holdtime command.

Field	Description
Type	Value can be one of the following: <ul style="list-style-type: none">• dynamic—NBMA address was obtained from NHRP Request packet.• static—NBMA address was statically configured.
Flags	Value can be one of the following: <ul style="list-style-type: none">• authoritative—Indicates that the NHRP information was obtained from the NHRP server or communication server that maintains the NBMA-to-IP addresss mapping for a particular destination.• implicit—Indicates that the information was learned not from an NHRP request generated from the local communication server, but from an NHRP packet being forwarded or from an NHRP request being received by the local communication server.• negative—For negative caching; indicates that the requested NBMA mapping could not be obtained.
NBMA address	Nonbroadcast, multiaccess address. The address format is appropriate for the type of network being used (for example, ATM, Ethernet, SMDS, multipoint tunnel).

Related Command
ip nhrp map

show ip nhrp traffic

To display Next Hop Resolution Protocol (NHRP) traffic statistics, use the **show ip nhrp traffic EXEC** command.

show ip nhrp traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip nhrp traffic** command:

```
cs# show ip nhrp traffic

Tunnel0
  request packets sent: 2
  request packets received: 4
  reply packets sent: 4
  reply packets received: 2
  register packets sent: 0
  register packets received: 0
  error packets sent: 0
  error packets received: 0
cs#
```

Table 18-14 describes the fields in the display.

Table 18-14 Show IP NHRP Traffic Field Descriptions

Field	Description
Tunnel 0	Interface type and number.
request packets sent	Number of NHRP request packets originated from this station.
request packets received	Number of NHRP request packets received by this station.
reply packets sent	Number of NHRP reply packets originated from this station.
reply packets received	Number of NHRP reply packets received by this station.
register packets sent	Number of NHRP register packets originated from this station. Currently, our communication servers do not send register packets, so this value is 0.
register packets received	Number of NHRP register packets received by this station. Currently, our communication servers do not send register packets, so this value is 0.
error packets sent	Number of NHRP error packets originated by this station.
error packets received	Number of NHRP error packets received by this station.

show ip redirects

To display the address of a default gateway (router) and the address of hosts for which a redirect has been received, use the **show ip redirects EXEC** command.

show ip redirects

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip redirects** command:

```
cs# show ip redirects
Default gateway is 160.89.80.29

Host          Gateway      Last Use    Total Uses  Interface
131.108.1.111 160.89.80.240 0:00        9   Ethernet0
128.95.1.4    160.89.80.240 0:00        4   Ethernet0
cs#
```

Related Command

ip redirects

show ip route

To display the entries in the routing table, use the **show ip route** EXEC command.

```
show ip route [address [mask]] | [protocol]
```

Syntax Description

<i>address</i>	(Optional) Address about which routing information should be displayed.
<i>mask</i>	(Optional) Argument for a subnet mask.
<i>protocol</i>	(Optional) Argument for a particular routing protocol, or static or connected .

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip route** command:

```
cs# show ip route 160.89.6.0
Routing entry for 160.89.6.0 (mask 255.255.255.0)
  Known via "connected", distance 0, metric 0 (connected)
  Tag 0
  Routing Descriptor Blocks:
  * directly connected, via Ethernet1
    Route metric is 0, traffic share count is 1
```

Table 18-14 describes the significant field shown in the display.

Table 18-14 Show IP Route Field Descriptions

Field	Description
Mask	Network mask associated with the route.
Connected	Routing protocol name, or connected or static .
Distance	Administrative distance.
Metric	Route metric that was either configured or learned from the particular route.
Routing Descriptor Blocks	Up to 4: Indicates the IP address of the next hop or the interface to which the particular route is connected.
*	Indicates the last path used when a packet was forwarded. It pertains only to the nonfast-switched packets. However, it does not indicate what path will be used next when forwarding a nonfast-switched packet except when the paths are equal cost.

show ip route summary

To display summary information about entries in the routing table, use the **show ip route summary EXEC** command.

show ip route summary

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip route summary** command:

```

cs# show ip route summary
Route Source      Networks  Subnets  Overhead  Memory (bytes)
connected         0         3         126       360
static            1         2         126       360
igrp 109          747       12        31878     91080
internal          3         0         0         360
Total             751       17        32130     92160
cs#
    
```

Table 18-15 describes the fields shown in the display:

Table 18-15 Show IP Route Summary Field Descriptions

Field	Description
Route Source	Routing protocol name, or connected , static , or internal . Internal—those routes that are in the primary routing table merely as markers to hold subnet routes. These routes are not owned by any routing protocol. There should be one of these internal routes for each subnetted network in the routing table.
Networks	The number of Class A, B, or C networks that are present in the routing table for each route source.
Subnets	The number of subnets that are present in the routing table for each route source, including host routes.
Overhead	Any additional memory involved in allocating the routes for the particular route source other than the memory specified under “Memory.”
Memory	The number of bytes allocated to maintain all the routes for the particular route source.

Related Command

show ip route

show ip tcp header-compression

To display statistics about TCP header compression, use the **show ip tcp header-compression EXEC** command.

show ip tcp header-compression

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip tcp header-compression** command:

```
cs# show ip tcp header-compression

TCP/IP header compression statistics:
Interface Serial1: (passive, compressing)
  Rcvd:   4060 total, 2891 compressed, 0 errors
         0 dropped, 1 buffer copies, 0 buffer failures
  Sent:   4284 total, 3224 compressed,
         105295 bytes saved, 661973 bytes sent
         1.15 efficiency improvement factor
  Connect: 16 slots, 1543 long searches, 2 misses, 99% hit ratio
         Five minute miss rate 0 misses/sec, 0 max misses/sec
```

Table 18-16 describes significant fields shown in the display.

Table 18-16 Show IP TCP Header-Compression Field Descriptions

Field	Description
Rcvd:	
total	Total number of TCP packets received.
compressed	Total number of TCP packets compressed.
errors	Unknown packets.
dropped	Number of packets dropped due to invalid compression.
buffer copies	Number of packets that had to be copied into bigger buffers for decompression.
buffer failures	Number of packets dropped due to a lack of buffers.
Sent:	
total	Total number of TCP packets sent.
compressed	Total number of TCP packets compressed.
bytes saved	Number of bytes reduced.
bytes sent	Number of bytes sent.
efficiency improvement factor	Improvement in line efficiency because of TCP header compression.

Field	Description
Connect:	
number of slots	Size of the cache.
long searches	Indicates the number of times the software had to look to find a match.
misses	Indicates the number of times a match could not be made. If your output shows a large miss rate, then the number of allowable simultaneous compression connections may be too small.
hit ratio	Percentage of times the software found a match and was able to compress the header.
Five minute miss rate	Calculates the miss rate over the previous 5 minutes for a longer-term (and more accurate) look at miss rate trends.
max misses/sec	Maximum value of the previous field.

Related Command

ip tcp header-compression

show ip traffic

To display statistics about IP traffic, use the **show ip traffic** EXEC command.

show ip traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip traffic** command:

```
cs# show ip traffic

IP statistics:
  Rcvd: 98 total, 98 local destination
        0 format errors, 0 checksum errors, 0 bad hop count
        0 unknown protocol, 0 not a gateway
        0 security failures, 0 bad options
  Frags:0 reassembled, 0 timeouts, 0 too big
        0 fragmented, 0 couldn't fragment
  Bcast:38 received, 52 sent
  Sent: 44 generated, 0 forwarded
        0 encapsulation failed, 0 no route
ICMP statistics:
  Rcvd: 0 checksum errors, 0 redirects, 0 unreachable, 0 echo
        0 echo reply, 0 mask requests, 0 mask replies, 0 quench
        0 parameter, 0 timestamp, 0 info request, 0 other
  Sent: 0 redirects, 3 unreachable, 0 echo, 0 echo reply
        0 mask requests, 0 mask replies, 0 quench, 0 timestamp
        0 info reply, 0 time exceeded, 0 parameter problem
UDP statistics:
  Rcvd: 56 total, 0 checksum errors, 55 no port
  Sent: 18 total, 0 forwarded broadcasts
TCP statistics:
  Rcvd: 0 total, 0 checksum errors, 0 no port
  Sent: 0 total
EGP statistics:
  Rcvd: 0 total, 0 format errors, 0 checksum errors, 0 no listener
  Sent: 0 total
IGRP statistics:
  Rcvd: 73 total, 0 checksum errors
  Sent: 26 total
HELLO statistics:
  Rcvd: 0 total, 0 checksum errors
  Sent: 0 total
ARP statistics:
  Rcvd: 20 requests, 17 replies, 0 reverse, 0 other
  Sent: 0 requests, 9 replies (0 proxy), 0 reverse
Probe statistics:
  Rcvd: 6 address requests, 0 address replies
  0 proxy name requests, 0 other
  Sent: 0 address requests, 4 address replies (0 proxy)
        0 proxy name replies
```

Table 18-17 describes significant fields shown in the display.

Table 18-17 Show IP Traffic Field Descriptions

Field	Description
format errors	A gross error in the packet format, such as an impossible Internet header length.
bad hop count	Occurs when a packet is discarded because its time-to-live (TTL) field was decremented to zero.
encapsulation failed	Usually indicates that the communication server had no ARP request entry and therefore did not send a datagram.
no route	Counted when the communication server discards a datagram it did not know how to route.
proxy name reply	Counted when the communication server sends an ARP or Probe Reply on behalf of another host. The display shows the number of probe proxy requests that have been received and the number of responses that have been sent.

term ip netmask-format

To specify the format in which network masks are displayed in **show** command output, use the **term ip netmask-format** EXEC command. To restore the default display format, use the **no** form of this command.

```
term ip netmask-format { bitcount | decimal | hexadecimal }  
term no ip netmask-format [bitcount | decimal | hexadecimal]
```

Syntax Description

bitcount	Addresses are followed by a slash and the total number of bits in the network mask. For example, 131.108.11.55/24 indicates that the network mask is 24 bits.
decimal	Network masks are displayed in dotted decimal notation (for example, 255.255.255.0).
hexadecimal	Network masks are displayed in hexadecimal format, as indicated by the leading 0X (for example, 0FFFFFFF00).

Default

Network masks are displayed in dotted decimal format.

Command Mode

EXEC

Usage Guidelines

IP uses a 32-bit mask that indicates which address bits belong to the network and subnetwork fields and which bits belong to the host field. This is called a network mask. By default, **show** commands display an IP address and then its network mask in dotted decimal notation. For example, a subnet would be displayed as 131.108.11.55 255.255.255.0.

However, you can specify that the network mask displays in hexadecimal format or bit count format. The hexadecimal format is commonly used on UNIX systems. The above example would be displayed as 131.108.11.55 0FFFFFFF00.

The bitcount format for displaying network masks is to append a slash (/) and the total number of bits in the network mask to the address itself. The above example would be displayed as 131.108.11.55/24.

Example

The following example specifies that network masks for the session be displayed in bitcount notation in the output of **show** commands:

```
term ip netmask-format bitcount
```

trace (user)

To discover the routes the router's packets follow when traveling to their destination, use the **trace** user EXEC command.

trace ip *destination*

Syntax Description

destination Destination address or host name on the command line. The default parameters for the appropriate protocol are assumed and the tracing action begins.

Command Mode

EXEC

Usage Guidelines

The **trace** command works by taking advantage of the error messages generated by communication servers when a datagram exceeds its time-to-live (TTL) value.

The **trace** command starts by sending probe datagrams with a TTL value of one. This causes the first communication server to discard the probe datagram and send back an error message. The **trace** command sends several probes at each TTL level and displays the round-trip time for each.

The **trace** command sends out one probe at a time. Each outgoing packet may result in one or two error messages. A *time exceeded* error message indicates that an intermediate communication server has seen and discarded the probe. A *destination unreachable* error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, **trace** prints an asterisk (*).

The **trace** command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with the escape sequence. By default, to invoke the escape sequence, press Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key.

Common Trace Problems

Due to bugs in the IP implementation of various hosts and communication servers, the IP **trace** command may behave in odd ways.

Not all destinations will respond correctly to a *probe* message by sending back an *ICMP port unreachable* message. A long sequence of TTL levels with only asterisks, terminating only when the maximum TTL has been reached, may indicate this problem.

There is a known problem with the way some hosts handle an *ICMP TTL exceeded* message. Some hosts generate an *ICMP* message but they reuse the TTL of the incoming packet. Since this is zero, the *ICMP* packets do not make it back. When you trace the path to such a host, you may see a set of TTL values with asterisks (*). Eventually the TTL gets high enough that the *ICMP* message can get back. For example, if the host is six hops away, **trace** will time out on responses 6 through 11.

Sample Display Showing Trace IP Routes

The following display shows sample IP **trace** output when a destination host name has been specified:

```
cs# trace ip ABA.NYC.mil
Type escape sequence to abort.
Tracing the route to ABA.NYC.mil (26.0.0.73)
 0 DEBRIS.CISCO.COM (131.108.1.6) 1000 msec 8 msec 4 msec
 1 BARNET-GW.CISCO.COM (131.108.16.2) 8 msec 8 msec 8 msec
 2 EXTERNAL-A-GATEWAY.STANFORD.EDU (192.42.110.225) 8 msec 4 msec 4 msec
 3 BB2.SU.BARNET.NET (131.119.254.6) 8 msec 8 msec 8 msec
 4 SU.ARC.BARNET.NET (131.119.3.8) 12 msec 12 msec 8 msec
 5 MOFFETT-FLD-MB.in.MIL (192.52.195.1) 216 msec 120 msec 132 msec
 6 ABA.NYC.mil (26.0.0.73) 412 msec 628 msec 664 msec
```

Table 18-18 describes the fields shown in the display.

Table 18-18 Trace Field Descriptions

Field	Description
1	Indicates the sequence number of the communication server in the path to the host.
DEBRIS.CISCO.COM	Host name of this communication server.
131.108.1.61	Internet address of this communication server.
1000 msec 8 msec 4 msec	Round-trip time for each of the three probes that are sent.

Table 18-19 describes the characters that can appear in **trace** output.

Table 18-19 IP Trace Text Characters

Char	Description
<i>mm</i> msec	For each node, the round-trip time in milliseconds for the specified number of probes.
*	The probe timed out.
?	Unknown packet type.
Q	Source quench.
P	Protocol unreachable.
N	Network unreachable.
U	Port unreachable.
H	Host unreachable.

Related Command

trace (privileged)

trace (privileged)

To discover the routes the router's packets follow when traveling to their destination, use the **trace** user EXEC command.

trace [*destination*]

Syntax Description

destination (Optional) Destination address or host name on the command line. The default parameters for the appropriate protocol are assumed and the tracing action begins.

Command Mode

Privileged EXEC

Usage Guidelines

The **trace** command works by taking advantage of the error messages generated by communication servers when a datagram exceeds its time-to-live (TTL) value.

The **trace** command starts by sending probe datagrams with a TTL value of one. This causes the first communication server to discard the probe datagram and send back an error message. The **trace** command sends several probes at each TTL level and displays the round-trip time for each.

The **trace** command sends out one probe at a time. Each outgoing packet may result in one or two error messages. A *time exceeded* error message indicates that an intermediate communication server has seen and discarded the probe. A *destination unreachable* error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, **trace** prints an asterisk (*).

The **trace** command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with the escape sequence. By default, to invoke the escape sequence, press Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key.

To use nondefault parameters and invoke an extended **trace** test, enter the command without a destination argument. You will be stepped through a dialog to select the desired parameters.

Common Trace Problems

Due to bugs in the IP implementation of various hosts and communication servers, the IP **trace** command may behave in odd ways.

Not all destinations will respond correctly to a *probe* message by sending back an *ICMP port unreachable* message. A long sequence of TTL levels with only asterisks, terminating only when the maximum TTL has been reached, may indicate this problem.

There is a known problem with the way some hosts handle an *ICMP TTL exceeded* message. Some hosts generate an *ICMP* message but they reuse the TTL of the incoming packet. Since this is zero, the *ICMP* packets do not make it back. When you trace the path to such a host, you may see a set of TTL values with asterisks (*). Eventually the TTL gets high enough that the *ICMP* message can get back. For example, if the host is six hops away, **trace** will time out on responses 6 through 11.

Sample Display Showing Trace IP Routes

The following display shows sample IP **trace** output when a destination host name has been specified:

```
cs# trace ABA.NYC.mil
Type escape sequence to abort.
Tracing the route to ABA.NYC.mil (26.0.0.73)
 1 DEBRIS.CISCO.COM (131.108.1.6) 1000 msec 8 msec 4 msec
 2 BARNET-GW.CISCO.COM (131.108.16.2) 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.BARNET.NET (131.119.254.6) 8 msec 8 msec 8 msec
 5 SU.ARC.BARNET.NET (131.119.3.8) 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
```

Table 18-20 describes the fields shown in the display.

Table 18-20 Trace Field Descriptions

Field	Description
1	Indicates the sequence number of the communication server in the path to the host.
DEBRIS.CISCO.COM	Host name of this communication server.
131.108.1.61	Internet address of this communication server.
1000 msec 8 msec 4 msec	Round-trip time for each of the three probes that are sent.

Sample Display Showing Extended IP Trace Dialog

The following display shows a sample **trace** session involving the extended dialog of the **trace** command:

```
cs# trace
Protocol [ip]:
Target IP address: mit.edu
Source address:
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to MIT.EDU (18.72.2.1)
 1 ICM-DC-2-V1.ICP.NET (192.108.209.17) 72 msec 72 msec 88 msec
 2 ICM-FIX-E-H0-T3.ICP.NET (192.157.65.122) 80 msec 128 msec 80 msec
 3 192.203.229.246 540 msec 88 msec 84 msec
 4 T3-2.WASHINGTON-DC-CNSS58.T3.ANS.NET (140.222.58.3) 84 msec 116 msec 88 msec
 5 T3-3.WASHINGTON-DC-CNSS56.T3.ANS.NET (140.222.56.4) 80 msec 132 msec 88 msec
 6 T3-0.NEW-YORK-CNSS32.T3.ANS.NET (140.222.32.1) 92 msec 132 msec 88 msec
 7 T3-0.HARTFORD-CNSS48.T3.ANS.NET (140.222.48.1) 88 msec 88 msec 88 msec
 8 T3-0.HARTFORD-CNSS49.T3.ANS.NET (140.222.49.1) 96 msec 104 msec 96 msec
 9 T3-0.ENSS134.T3.ANS.NET (140.222.134.1) 92 msec 128 msec 92 msec
10 W91-CISCO-EXTERNAL-FDDI.MIT.EDU (192.233.33.1) 92 msec 92 msec 112 msec
11 E40-RTR-FDDI.MIT.EDU (18.168.0.2) 92 msec 120 msec 96 msec
12 MIT.EDU (18.72.2.1) 96 msec 92 msec 96 msec
```

Table 18-21 describes the fields that are unique to the extended trace sequence, as shown in the display.

Table 18-21 Trace Field Descriptions

Field	Description
Target IP address	You must enter a host name or an IP address. There is no default.
Source address	One of the interface addresses of the communication server to use as a source address for the probes. The communication server will normally pick what it feels is the best source address to use.
Numeric display	The default is to have both a symbolic and numeric display; however, you can suppress the symbolic display.
Timeout in seconds	The number of seconds to wait for a response to a probe packet. The default is 3 seconds.
Probe count	The number of probes to be sent at each TTL level. The default count is 3.
Minimum Time to Live [1]	The TTL value for the first probes. The default is 1, but it can be set to a higher value to suppress the display of known hops.
Maximum Time to Live [30]	The largest TTL value that can be used. The default is 30. The trace command terminates when the destination is reached or when this value is reached.
Port Number	The destination port used by the UDP probe messages. The default is 33434.
Loose, Strict, Record, Timestamp, Verbose	IP header options. You may specify any combination. The trace command issues prompts for the required fields. Note that trace will place the requested options in each probe; however, there is no guarantee that all communication servers (or end nodes) will process the options.
Loose Source Routing	Allows you to specify a list of nodes that must be traversed when going to the destination.
Strict Source Routing	Allows you to specify a list of nodes that must be the only nodes traversed when going to the destination.
Record	Allows you to specify the number of hops to leave room for.
Timestamp	Allows you to specify the number of time stamps to leave room for.
Verbose	If you select any option, the verbose mode is automatically selected and trace prints the contents of the option field in any incoming packets. You can prevent verbose mode by selecting it again, toggling its current setting.

Table 18-22 describes the characters that can appear in **trace** output.

Table 18-22 IP Trace Text Characters

Char	Description
<i>nn</i> msec	For each node, the round-trip time in milliseconds for the specified number of probes.
*	The probe timed out.
?	Unknown packet type.
Q	Source quench.
P	Protocol unreachable.

Char	Description
N	Network unreachable.
U	Port unreachable.
H	Host unreachable.

Related Command

trace (user)

