CHAPTER 16

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 *Communication Server Configuration Guide*.

access-class

Use the **access-class** line configuration command to restrict incoming and outgoing connections between a particular virtual terminal line (into a Cisco device) and the addresses in an access list. The **no access-class** command removes access restrictions on the line for the specified connections.

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

Syntax Description

access-list-number	Integer from 1 through 99 that identifies a specific access list of Internet addresses.
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

None

Command Mode

Line configuration command

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)

Use the **access-list** global configuration command to create or remove an access list and control access to it. Use the **no access-list** command to delete the entire access list.

access-list *access-list-number* {**permit** | **deny**} *source* [*source-mask*]
no access-list *access-list-number*

Syntax Description

access-list-number	Integer from 1 through 99 that you assign to identify one or more permit/deny conditions as an access list. Access list 0 (zero) is predefined; it permits any address and is the default access list for all interfaces.
permit	Permits access for matching conditions.
deny	Denies access to matching conditions.
source	Compares the source address being tested to this value. It is a 32-bit quantity written in dotted-decimal format. See the examples later in this section.
source-mask	32-bit quantity written in dotted-decimal format. Address bits corresponding to wildcard mask bits set to 1 are ignored in comparisons; address bits corresponding to wildcard mask bits set to zero are used in comparisons. See the examples later in this section.

Default

The access list defaults to an implicit deny statement for everything that has not been permitted.

Command Mode

Global configuration

Usage Guidelines

Plan your access conditions carefully and be aware of the implicit deny.

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

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

Examples

The following example of a standard access list allows access for only those hosts on the three specified networks. It assumes that subnetting is not used; the masks apply to the host portions of the network addresses. Any hosts 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 128.88.1.0  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 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 Command

show access-lists

access-list (extended)

Use the extended **access-list** global configuration command to create or remove an extended access list. Use the **no access-list** command to delete the entire extended access list.

access-list *access-list-number* {**permit** | **deny**} *protocol source source-mask destination destination-mask* [*operator operand*] [**established**]

no access-list *access-list-number*

Syntax Description

access-list-number	Integer from 100 through 199 that you assign to identify one or more extended permit/deny conditions as an extended access list. Note that a list number in the range 100 through 199 distinguishes an extended access list from a standard access list.
permit	Permits access to matching conditions.
deny	Denies access to matching conditions.
protocol	One of the following protocols: **ip**, **tcp**, **udp**, **icmp**, **igmp**, **gre**, or **igrp** or an integer in the range of 0 through 255 representing an IP protocol number. Use the keyword **ip** to match any Internet protocol, including TCP, UDP, and ICMP.
source	Internet source address in dotted-decimal format. Used in conjunction with source masks.
source-mask	Mask of source address bits in dotted-decimal format. The *source* and *source-mask* arguments are used to match the source address of a packet.
destination	Internet destination address in dotted-decimal format. Used in conjunction with destination masks.
destination-mask	Mask of destination address bits in dotted-decimal format. The destination and destination mask arguments are used to match the destination address of a packet.
operator	(Optional.) Compares destination ports. Note that the **ip** and **icmp** protocol keywords do not allow port distinctions. Possible operands include **lt** (less than), **gt** (greater than), **eq** (equal), and **neq** (not equal).
operand	(Optional.) Decimal destination port. Note that the **ip** and **icmp** protocol keywords do not allow port distinctions.
established	(Optional.) For the TCP protocol only: to indicate an established connection. A match occurs if the TCP datagram has the ACK or RST bits set. The nonmatching case is that of the initial TCP datagram to form a connection.

Default

An extended access list defaults to an implicit deny statement for everything that has not been permitted.

Command Mode

Global configuration

Usage Guidelines

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

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.

Example

In the following example, the Ethernet network is 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 ethernet 0
ip access-group 102
```

Related Commands

ip access-group
show access-lists

arp (global)

Use the **arp** global configuration command to install a permanent entry in the ARP cache. The communication server uses this entry to translate 32-bit Internet Protocol addresses into 48-bit hardware addresses. Use the **no arp** command to remove the specified entry from the ARP cache.

arp *internet-address hardware-address type* [**alias**]
no arp *internet-address hardware-address type* [**alias**]

Syntax Description

internet-address	Internet address in dotted-decimal format corresponding to the local data link address.
hardware-address	Local data link address (a 48-bit address).
type	Encapsulation description. This is typically the **arpa** keyword for Ethernet.
alias	(Optional.) Indicates that the communication server should respond to ARP requests as if it were the owner of the specified address.

Default

None

Command Mode

Global configuration

Usage Guidelines

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)

Use the **arp** interface configuration command to control the interface-specific handling of IP address resolution into 48-bit Ethernet and Token Ring hardware addresses. Use the **no arp** command to selectively disable the specified interface encapsulation type.

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

arpa

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

Use the **arp timeout** interface configuration command to control the number of seconds an ARP cache entry will stay in the cache. Use the **no arp timeout** command to restore the default value.

arp timeout *seconds*
no arp timeout *seconds*

Syntax Description

seconds	Value used to age an ARP cache entry related to that interface. A value of 0 (zero) seconds sets no timeout; then the cache entries are never cleared.

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

show interfaces

clear host

Use the **clear host** EXEC command to remove one or all entries from the host name-and-address cache.

clear host {*name* | ***}

Syntax Description

name	Particular host entry to remove.
*****	Remove 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 Command

show hosts

clear ip accounting

Use the **clear ip accounting** EXEC command to clear the active database when IP accounting is enabled. Use the **clear ip accounting checkpoint** command to clear the checkpointed database when IP accounting is enabled.

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 route

Use the **clear ip route** EXEC command to remove one or more routes from the IP routing table.

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

Syntax Description

network	Network or subnet address to remove.
mask	(Optional.) Subnet address to remove.
*	Remove 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
```

ip access-group

Use the **ip access-group** interface configuration command to control access to an interface. Use the **no ip access-group** command to remove the specified access group.

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

Syntax Description

access-list-number	Access list number from 1 through 199.
in	Filter on inbound packets.
out	Filter on outbound packets. Default if **in** or **out** not specified.

Default

None

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 0.

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

Related Commands

access-list (extended)
show access-lists

ip accounting

Use the **ip accounting** interface configuration command to enable IP accounting on an interface. Use the **no ip accounting** command to disable IP accounting.

ip accounting
no ip accounting

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP accounting is enabled on a per-interface basis. The IP accounting support records the number of bytes and packets switched through the system on a source and destination IP address basis. 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.

Example

The following example enables IP accounting on interface Ethernet 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

Use the **ip accounting-list** global configuration command to specify a set of filters to control the hosts for which IP accounting information is kept. Use the **no ip accounting-list** command with the appropriate argument to remove this function.

ip accounting-list *ip-address mask*
no ip accounting-list *ip-address mask*

Syntax Description

ip-address	IP address in dotted-decimal format.
mask	IP mask.

Default

None

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

Use the **ip accounting-threshold** global configuration command to enable IP accounting for transit traffic outbound on an interface. Use the **no ip accounting-threshold** command to restore the default.

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, preventing IP accounting from possibly consuming all available free memory.

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.

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

Use the **ip accounting-transits** global configuration command to control the number of transit records that will be stored in the IP accounting database. Use the **no ip accounting-transits** command to remove this function, resetting the value to the default.

ip accounting-transits *count*
no ip accounting-transits

Syntax Description

count	Number of transit records that will be stored 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-transit 100
```

Related Commands

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

ip address

Use the **ip address** interface configuration command to set an IP address for an interface. Use the **no ip address** command to remove the specified address.

ip address *IP-address mask*
no ip address *IP-address mask*

Syntax Description

IP-address	IP address.
mask	Mask for the associated IP subnet.

Default

None

Command Mode

Interface configuration

Usage Guidelines

Hosts can determine subnet masks using the Internet Control Message Protocol (ICMP) *Mask Request* message. Routers 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 0.

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

ip address secondary

Use the **ip address secondary** interface configuration command to set multiple IP addresses for an interface. Use the **no ip address secondary** command to remove the specified addresses.

ip address *IP-address mask* **secondary**
no ip address *IP-address mask* **secondary**

Syntax Description

IP-address	IP address.
mask	Mask for the associated IP subnet.
secondary	Used to specify additional IP addresses.

Default

None

Command Mode

Interface configuration

Usage Guidelines

Hosts can determine subnet masks using the Internet Control Message Protocol (ICMP) *Mask Request* message. Routers 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 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

Use the **ip broadcast-address** interface configuration command to define a broadcast address for an interface. Use the **no ip broadcast-address** command to restore the IP broadcast address to the default.

ip broadcast-address [*address*]
no ip broadcast-address [*address*]

Syntax Description

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 default-gateway

Use the **ip default-gateway** global configuration command to define a default gateway (communication server) when IP routing is disabled. Use the **no ip default-gateway** command to disable this function.

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

Syntax Description

address	Internet 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 Internet 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

Use the **ip directed-broadcast** interface configuration command to enable directed broadcast to physical broadcast translation on an interface. Use the **no ip directed-broadcast** command to disable directed broadcast to physical broadcast translation on an interface.

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 will be forwarded.

Default

Enabled, with no list specified

Command Mode

Interface configuration

Usage Guidelines

By default, 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 interface Ethernet 0:

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

Related Command

ip forward-protocol

ip domain-list

Use the **ip domain-list** global configuration command to define a list of default domain names to complete unqualified host names. Use the **no ip domain-list** command with the appropriate argument to delete a name from the list.

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

None

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. 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

Use the **ip domain-lookup** global configuration command to enable the IP Domain Name System-based host name-to-address translation. Use the **no ip domain-lookup** command to disable the Domain Name System.

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

Use the **ip domain-name** global configuration command 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 **no ip domain-name** command to disable the use of the Domain Name System.

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

Use the **ip forward-protocol** global configuration command to specify which protocols and ports the communication server will forward. Use the **no ip forward-protocol** command (with the appropriate keyword and argument) to remove the protocol/port. Specifying just the protocol, without the port, disables all flooding for that protocol.

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

Syntax Description

udp	Forward UDP datagrams. See below for a list of datagrams forwarded by default.
nd	Forward Network Disk (ND) datagrams. This protocol is used by older diskless SUN workstations. See below for a list of datagrams forwarded by default.
port	(Optional.) Destination port that controls which UDP services are forwarded.

Default

If a helper address is specified and UDP forwarding is enabled, the following datagrams 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 (port 67)
- TACACS service (port 49)

Command Mode

Global configuration

Example

The following example uses the **ip forward-protocol** command to specify forwarding of UDP only, then defines a helper address.

```
ip forward-protocol udp
!
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 gdp igrp

Use the **ip gdp igrp** interface configuration command to configure the communication server discovery feature using the Cisco IGRP routing protocol. Use the **no ip gdp igrp** command to disable this feature.

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 the Ethernet 1 interface:

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

ip gdp irdp

Use the **ip gdp irdp** interface configuration command to configure the communication server discovery feature using the ICMP Router Discovery Protocol (IRDP). Use the **no ip gdp irdp** command to disable this feature.

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 0 interface:

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

ip gdp rip

Use the **ip gdp rip** interface configuration command to configure the communication server discovery feature using the RIP routing protocol. Use the **no ip gdp rip** command to disable this feature.

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 the Ethernet 1 interface:

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

ip helper-address

Use the **ip helper-address** interface configuration command to tell the communication server to forward UDP broadcasts, including BootP, received on an interface. Use the **no ip helper-address** command to disable the forwarding of broadcast packets to specific addresses.

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.

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

Use the **ip host** global configuration command to define a static host name-to-address mapping in the host cache. Use the **no ip host** command to remove the name-to-address mapping.

ip host *name* [*TCP-port-number*] *address1* [*address2...address8*]
no ip host *name address*

Syntax Description

name	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.
TCP-port-number	(Optional.) TCP port number—Telnet by default (port 23).
address	Associated IP address. Up to eight addresses can be bound 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 host-routing

Use the **ip host-routing** command to configure your communication server to act as a terminal server.

ip host-routing
no ip host-routing

Syntax Description

There are no arguments or keywords for this command.

Default

Command Mode

Global configuration

Usage Guidelines

The functionality of this command compares to the functionality of the **ip routing** command as follows:

- **ip routing**—Run the configured routing protocols. If communication servers are not configured, do not send packets.
- **no ip routing**—Do not run routing protocols. If the destination is not on the same subnet, use ARP and depend on proxies.
- **ip host-routing**—Do not run routing protocols. If you are not on the same subnet, use ARP and depend on proxies. This command allows IP routing between the SLIP and PPP hosts attached to the communication server but uses host routing methods to send packets to devices and networks that are not.

Example

The following example uses the **ip host-routing** command to configure the communication server to act as a terminal server:

```
ip host-routing
```

Related Commands

ip routing

ip hp-host

Use the **ip hp-host** global configuration command to enter the host name of an HP host to be used for HP Probe Proxy service into the host table. Use the **no ip hp-host** command with the appropriate arguments to remove the host name.

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

Syntax Description

hostname	Name of the host.
ip-address	IP address of the host.

Default

None

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

Use the **ip mask-reply** interface configuration command to tell the communication server to respond to ICMP mask requests by sending ICMP Mask Reply messages. Use the **no ip mask-reply** command to disable this function.

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 interface Ethernet 0.

```
interface ethernet 0
ip mask-reply
```

ip mtu

Use the **ip mtu** interface configuration command to set the maximum transmission unit (MTU) size of IP packets sent on an interface. Use the **no ip mtu** command to restore the default.

ip mtu *bytes*
no ip mtu

Syntax Description

bytes	IP MTU in bytes.

Default

Minimum is 128 bytes; maximum depends on interface medium type

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.

Also, 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

Use the **ip name-server** global configuration command to specify the address of one or more name servers to use for name and address resolution. Use the **no ip name-server** command to remove the addresses specified and restore the default.

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

None

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
```

Related Commands

ip domain-lookup
ip domain-name

ip probe proxy

Use the **ip probe proxy** interface configuration command to enable the HP Probe Proxy support that allows a communication server to respond to HP Probe Proxy Name requests. Use the **no ip probe proxy** command to disable HP Probe Proxy.

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

Use the **ip proxy-arp** interface configuration command to enable proxy ARP on an interface. Use the **no ip proxy-arp** command to disable proxy ARP on the interface.

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 interface Ethernet 0.

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

ip redirects

Use the **ip redirects** interface configuration command to enable the sending of redirect messages if the communication server is forced to resend a packet through the same interface on which it was received. Use the **no ip redirects** command to disable the sending of redirect messages.

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 interface Ethernet 0.

```
interface ethernet 0
ip redirects
```

Related Command

show ip redirects

ip routing

Use the **ip routing** global configuration command to enable IP routing. Use the **no ip routing** command to disable IP routing for the communication server.

ip routing
no ip routing

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Example

The following example shows how to enable IP routing.

```
ip routing
```

ip security add

Use the **ip security add** interface configuration command to add a basic security option to all outgoing packets. Use the **no ip security add** command to disable the adding of a basic security option to all outgoing packets.

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). Other wise, 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 interface Ethernet 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 dedicated

Use the **ip security dedicated** interface configuration command to set the requested level of classification and authority on the interface. Use the **no ip security dedicated** command to reset the interface to the default classification and authorities.

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

Syntax Description

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

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 1-1.

Table 1-1 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 Defense Communications Agency (DCA). The authority keywords and their corresponding bit patterns are shown in Table 1-2.

Table 1-2 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 extended-allowed

Use the **ip security extended-allowed** interface configuration command to accept packets on an interface that has an extended security option present. Packets containing extended security options are rejected. Use the **no ip security extended-allowed** command to restore the default.

ip security extended-allowed
no ip security extended-allowed

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Example

The following example allows interface Ethernet 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

Use the **ip security first** interface configuration command to prioritize the presence of security options on a packet. Use the **no ip security first** command to turn off this function.

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 interface Ethernet 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

Use the **ip security ignore-authorities** interface configuration command to causes the communication server to ignore the authorities field of all incoming packets. Use the **no ip security ignore-authorities** command to turn off this function.

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 interface Ethernet 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

Use the **ip security implicit-labelling** interface configuration command to force the communication server to accept packets on the interface, even if they do not include a security option. Use the **no ip security implicit-labelling** command to disable this function.

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

Syntax Description

level	(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 1-1 (see the **ip security dedicated** interface configuration command).
authority	(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 1-2 (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

Use the **ip security multilevel** interface configuration command to set the interface to the requested range of classifications and authorities. All traffic entering or leaving the system must have a security option that falls within this range.

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

Syntax Description

level1	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 1-1 (see the **ip security dedicated** command).
authority1	(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 1-2 (see the **ip security dedicated** command).
to	Separates the range of classifications and authorities.
level2	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 1-1 (see the **ip security dedicated** command).
authority2	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 1-2 (see the **ip security dedicated** command).

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

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

Use the **ip security reserved-allowed** interface configuration command to treat as valid any packets that have Reserved1 through Reserved4 security levels. Use the **no ip security reserved-allowed** command to disable this feature.

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 interface Ethernet 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

Use the **ip security strip** interface configuration command to remove any basic security option on outgoing packets on an interface. Use the **no ip security strip** command to disable this function.

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 interface Ethernet 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

Use the **ip source-route** global configuration command to allow the communication server to handle IP datagrams with source routing header options. Use the **no ip source-route** command to cause the system to discard any IP datagram containing a source-route option.

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

Use the **ip subnet-zero** global configuration command to enable use of subnet zero for interface addresses and routing updates. Hence, it provides the ability to configure and route to subnet-zero subnets. Use the **no ip subnet-zero** command to restore the default.

ip subnet-zero
no ip subnet-zero

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

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

Use the **ip tcp compression-connections** interface configuration command to specify the total number of header compression connections that can exist on an interface. Use the **no ip tcp compression-connections** command to restore the default.

ip tcp compression-connections *number*
no ip tcp compression-connections *number*

Syntax Description

number	Number of connections the cache will support; *number* can vary between 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

Use the **ip tcp header-compression** interface configuration command to enable TCP header compression. Use the **no ip tcp header-compression** command to disable compression.

ip tcp header-compression [**passive**]
no ip tcp header-compression [**passive**]

Syntax Description

passive	(Optional.) Outgoing TCP packets are compressed 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 only supported on serial lines using HDLC 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.

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

Use the **ip tcp synwait-time** global configuration command to set a specified period of time the communication server will wait to attempt to establish a TCP connection before it times out. The **no ip tcp synwait-time** command restores the default.

ip tcp synwait-time *seconds*
no ip tcp synwait-time *seconds*

Syntax Description

seconds	Number of seconds the communication server waits to attempt to establish a TCP connection. Use any value between 5 and 300 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

Use the **ip unnumbered** interface configuration command to enable IP processing on a serial interface without assigning an explicit IP address to the interface. Use the **no ip unnumbered** command to disable the IP processing on the interface.

ip unnumbered *interface-name*
no ip unnumbered *interface-name*

Syntax Description

interface-name	Name of another interface on which the communication server has an assigned IP address. This *interface-name* cannot be another unnumbered interface.

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 *interface-name* argument 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 unreachables

Use the **ip unreachables** interface configuration command to enable the generation of ICMP Unreachable messages on a specified interface. Use the **no ip unreachables** command to disable this function.

ip unreachables
no ip unreachables

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.

Example

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

```
interface ethernet 0
ip unreachables
```

ping (privileged)

Use the **ping** (IP packet internet groper function) privileged EXEC command to send ICMP *Echo* messages to check host reachability and network connectivity. If the communication server receives an ICMP *Echo* message, it sends an ICMP *Echo Reply* message to the source of the ICMP *Echo* message.

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

Syntax Description

protocol	(Optional.) Protocol keyword. IP is the default.
host	Host name of system to ping.
address	IP address of system to ping.

Command Mode

Privileged EXEC

Usage Guidelines

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.

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 1-3 describes the test characters that the ping facility sends.

Table 1-3 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.

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 1-4 describes significant fields shown in the display.

Table 1-4 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.
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.

Field	Description
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 *Parameter Problem* message to the source of the packet and discards the packet. The Internet header options follow: • 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]:
! The “r” specifies the Record Route option
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
```

Following is 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
```

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 (unprivileged)

ping (unprivileged)

Use the **ping** (IP packet internet groper function) unprivileged EXEC command to send ICMP *Echo* messages to check host reachability and network connectivity. If the communication server receives an ICMP *Echo* message, it sends an ICMP *Echo Reply* message to the source of the ICMP *Echo* message.

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

Syntax Description

protocol	(Optional.) Protocol keyword. IP is the default.
host	Host name of system to ping.
address	IP address of system to ping.

Command Mode

EXEC

Usage Guidelines

The unprivileged 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 unprivileged pings.

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

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 1-5 describes the test characters that the ping facility sends.

Table 1-5 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.

Char	Description
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:

```
communication server> 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:

```
communication server> 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)

show access-lists

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

show access-lists

Syntax Description

This command has no arguments or keywords.

Command Mode

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 *Communication Server Configuration Guide*.

Related Command

access-list

show arp

Use the **show arp** EXEC command to display the entries in the ARP table for the communication server.

show arp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

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

```
cs# show arp

Protocol   Address          Age (min)   Hardware Addr    Type    Interface
```

Table 1-6 describes significant fields shown in the first line of output in the display.

Table 1-6 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: • ARPA • SNAP • ETLK (EtherTalk) • SMDS
Interface	Indicates the interface associated with this network address.

show hosts

Use the **show hosts** EXEC command 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.

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
Hame/address lookup uses domain service
Hame 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 1-7 describes significant fields shown in the display.

Table 1-7 Show Hosts Field Descriptions

Field	Description
Flag	A temp 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 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 accounting

Use the **show ip accounting** EXEC command to display the active accounting or checkpointed database.

show ip accounting [**checkpoint**]

Syntax Description

checkpoint	(Optional.) Indicates that the checkpointed database should be displayed.

Command Mode

EXEC

Sample Display

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 1-8 describes significant fields shown in the display.

Table 1-8 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.

Related Commands

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

show ip aliases

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

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

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

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      Ethernet0
```

Table 1-9 describes significant fields shown in the display.

Table 1-9 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: • ARPA—Ethernet • SNAP—RFC 1042 • ISO1—IEEE 802.3
Interface	Interface to which this address mapping has been assigned.

show ip interface

Use the **show ip interface** EXEC command to display the usability status of interfaces.

show ip interface [*interface unit*]

Syntax Description

interface unit	(Optional.) Used to display information for a particular interface. For example, e 0 specifies the first Ethernet interface; e 1 specifies the second Ethernet interface. You must specify both the interface type and unit 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.

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 unreachables are always sent
    ICMP mask replies are never sent
    Gateway Discovery is disabled
    IP accounting is disabled
    TCP/IP header compression is disabled
    Probe proxy name replies are disabled
```

Table 1-10 describes significant fields shown in the display.

Table 1-10 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 unreachables	Specifies whether unreachable messages will be sent on this interface.
ICMP mask replies	Specifies whether mask replies will be sent on this interface.
Gateway Discovery	Specifies whether the discovery process has been enabled for this interface. It is generally disabled on serial interfaces, such as this one.
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

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

show ip masks *address*

Syntax Description

address	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 redirects

Use the **show ip redirects** EXEC command to display the address of a default gateway (communication server).

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

Use the **show ip route** EXEC command to display the current state of the routing table.

show ip route [*address* [*mask*] | *protocol* | **summary**] [*process-id*]]

Syntax Description

address	(Optional.) Address about which routing information should be displayed.
mask	(Optional.) Argument for a subnet mask.
protocol	(Optional.) Argument for a particular routing protocol, or **static** or **connected**.
summary	Summary information about all routes.
process-id	(Optional.) Identifies the particular routing protocol process.

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

cs#
```

Table 1-11 describes the significant field shown in the display.

Table 1-11 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.
*	Round-robin pointer. It indicates the last path used when a packet was forwarded. The pointer applies to nonfast-switched packets only. The asterisk gives no indication as to which path will be used next when forwarding a nonfast-switched packet except when the paths are of equal cost.

show ip tcp header-compression

Use the **show ip tcp header-compression** EXEC command to display statistics on TCP header compression.

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 1-12 describes significant fields shown in the display.

Table 1-12 Show IP TCP Header Compression

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.
0 max misses/sec	Maximum value of the previous field.

Related Command

ip tcp header-compression

show ip traffic

Use the **show ip traffic** EXEC command to display IP protocol statistics.

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 1-13 describes significant fields shown in the display.

Table 1-13 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.

trace (privileged)

Use the **trace** privileged EXEC command to discover the routes the communication server's packets will actually take when traveling to their destination.

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 BARRNET-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.BARRNET.NET (131.119.254.6) 8 msec 8 msec 8 msec
  5 SU.ARC.BARRNET.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 1-14 describes the fields shown in the display.

Table 1-14 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
12 MIT.EDU (18.72.2.1) 96 msec 92 msec 96 msec
```

Table 1-15 describes the fields that are unique to the extended trace sequence, as shown in the display.

Table 1-15 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 1-16 describes the characters that can appear in **trace** output.

Table 1-16 IP Trace Text Characters

Char	Description
nn 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 (unprivileged)

trace (user)

Use the **trace** EXEC command to discover the IP routes the communication server's packets will actually take when traveling to their destination.

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)
  1 DEBRIS.CISCO.COM (131.108.1.6) 1000 msec 8 msec 4 msec
  2 BARRNET-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.BARRNET.NET (131.119.254.6) 8 msec 8 msec 8 msec
  5 SU.ARC.BARRNET.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 1-17 describes the fields shown in the display.

Table 1-17 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 1-18 describes the characters that can appear in **trace** output.

Table 1-18 IP Trace Text Characters

Char	Description
nn 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)