



# IP Addressing 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 the addressing of IP networks. For IP addressing configuration information and examples, refer to the “Configuring IP Addressing” chapter of the *Network Protocols Configuration Guide, Part 1*.

## arp (global)

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

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

### Syntax Description

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

### Default

No entries are permanently installed in the ARP cache.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Cisco IOS software uses ARP cache entries to translate 32-bit IP addresses into 48-bit hardware addresses.

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

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

### Example

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

```
arp 192.31.7.19 0800.0900.1834 arpa
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear arp-cache**

---

## arp (interface)

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

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

### Syntax Description

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

### Default

Standard Ethernet-style ARP

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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 Cisco IOS software would send three (two for **probe** and one for **arpa**) packets each time it needed to discover a Media Access Control (MAC) address.

The **arp probe** command allows the software 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 software 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

You can use the master indexes or search online to find documentation of related commands.

**clear arp-cache**

**show interfaces**

## arp timeout

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

```
arp timeout seconds  
no arp timeout seconds
```

### Syntax Description

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

### Default

14400 seconds (4 hours)

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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 sets the ARP timeout to 12000 seconds to allow entries to time out more quickly than the default:

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**show interfaces**

## clear arp-cache

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

**clear arp-cache**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in IOS Release 10.0.

### Example

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

```
clear arp-cache
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**arp (global)**

**arp (interface)**

## clear host

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

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

### Syntax Description

*name* Particular host entry to remove.

\* Removes all entries.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in IOS Release 10.0.

The host name entries will not be removed from nonvolatile random-access memory (NVRAM), but will be cleared in running memory.

### Example

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

```
clear host *
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip host**  
**show hosts**

## clear ip nat translation

To clear dynamic Network Address Translation (NAT) translations from the translation table, use the **clear ip nat translation EXEC** command.

```
clear ip nat translation { * | [inside global-ip local-ip] [outside local-ip global-ip]}  
clear ip nat translation protocol inside global-ip global-port local-ip local-port [outside  
local-ip global-ip]
```

### Syntax Description

<b>*</b>	Clears all dynamic translations.
<b>inside</b>	Clears the inside translations containing the specified <i>global-ip</i> and <i>local-ip</i> addresses.
<i>global-ip</i>	When used without the arguments <i>protocol</i> , <i>global-port</i> , and <i>local-port</i> , clears a simple translation that also contains the specified <i>local-ip</i> address. When used with the arguments <i>protocol</i> , <i>global-port</i> , and <i>local-port</i> , clears an extended translation.
<i>local-ip</i>	(Optional) Clears an entry that contains this local IP address and the specified <i>global-ip</i> address.
<b>outside</b>	Clears the outside translations containing the specified <i>global-ip</i> and <i>local-ip</i> addresses.
<i>protocol</i>	(Optional) Clears an entry that contains this protocol and the specified <i>global-ip</i> address, <i>local-ip</i> address, <i>global-port</i> , and <i>local-port</i> .
<i>global-port</i>	(Optional) Clears an entry that contains this <i>global-port</i> and the specified <i>protocol</i> , <i>global-ip</i> address, <i>local-ip</i> address, and <i>local-port</i> .
<i>local-port</i>	(Optional) Clears an entry that contains this <i>local-port</i> and the specified <i>protocol</i> , <i>global-ip</i> address, <i>local-ip</i> address, and <i>global-port</i> .

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Use this command to clear entries from the translation table before they time out.

### Example

The following example shows the NAT entries before and after the UDP entry being cleared:

```
Router# show ip nat translation  
Pro Inside global      Inside local      Outside local      Outside global  
udp 171.69.233.209:1220 192.168.1.95:1220 171.69.2.132:53    171.69.2.132:53  
tcp 171.69.233.209:11012 192.168.1.89:11012 171.69.1.220:23    171.69.1.220:23  
tcp 171.69.233.209:1067 192.168.1.95:1067 171.69.1.161:23    171.69.1.161:23
```

```
Router# clear ip nat translation udp inside 171.69.233.209 1220 192.168.1.95 1220
171.69.2.132 53 171.69.2.132 53
```

```
Router# show ip nat translation
Pro Inside global      Inside local      Outside local      Outside global
tcp 171.69.233.209:11012 192.168.1.89:11012 171.69.1.220:23    171.69.1.220:23
tcp 171.69.233.209:1067 192.168.1.95:1067 171.69.1.161:23    171.69.1.161:23
```

## Related Commands

You can use the master indexes or search online to find documentation of related commands.

- ip nat**
- ip nat inside destination**
- ip nat inside source**
- ip nat outside source**
- ip nat pool**
- ip nat translation**
- show ip nat statistics**
- show ip nat translations**

## clear ip nhrp

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

**clear ip nhrp**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

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

### Example

The following example clears all dynamic entries from the NHRP cache for the interface:

```
clear ip nhrp
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**show ip nhrp**

## clear ip route

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

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

### Syntax Description

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

### Default

All entries are removed.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### 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 address

To set a primary or secondary IP address for an interface, use the **ip address** interface configuration command. To remove an IP address or disable IP processing, use the **no** form of this command.

```
ip address ip-address mask [secondary]  
no ip address ip-address mask [secondary]
```

### Syntax Description

<i>ip-address</i>	IP address.
<i>mask</i>	Mask for the associated IP subnet.
<b>secondary</b>	(Optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.

### Default

No IP address is defined for the interface.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

An interface can have one primary IP address and multiple secondary IP addresses. Packets generated by the Cisco IOS software always use the primary IP address. Therefore, all routers and access servers on a segment should share the same primary network number.

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 software detects another host using one of its IP addresses, it will print an error message on the console.

The optional keyword **secondary** allows you to specify an unlimited number of secondary addresses. Secondary addresses are treated like primary addresses, except the system never generates datagrams other than routing updates with secondary source addresses. IP broadcasts and ARP requests are handled properly, as are interface routes in the IP routing table.

Secondary IP addresses can be used in a variety of situations. The following are the most common applications:

- There may not be enough host addresses for a particular network segment. For example, your subnetting allows up to 254 hosts per logical subnet, but on one physical subnet you need to have 300 host addresses. Using secondary IP addresses on the routers or access servers allows you to have two logical subnets using one physical subnet.
- Many older networks were built using Level 2 bridges. The judicious use of secondary addresses can aid in the transition to a subnetted, router-based network. Routers on an older, bridged segment can be easily made aware that there are many subnets on that segment.

- Two subnets of a single network might otherwise be separated by another network. This situation is not permitted when subnets are in use. In these instances, the first network is *extended*, or layered on top of the second network using secondary addresses.

---

**Note** If any router on a network segment uses a secondary address, all other devices on that same segment must also use a secondary address from the same network or subnet. Inconsistent use of secondary addresses on a network segment can very quickly cause routing loops.

---

---

**Note** When you are routing Open Shortest Path First (OSPF), ensure that all secondary addresses of an interface fall into the same OSPF area as the primary addresses.

---

To transparently bridge IP on an interface, you must do two things:

- Disable IP routing (specify **no ip routing**).
- Add the interface to a bridge group. (See the **bridge-group** command.)

To concurrently route and transparently bridge IP on an interface, see the **bridge crb** command.

### Example

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge crb**

**bridge-group**

## ip broadcast-address

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

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

### Syntax Description

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

### Default

Default address: 255.255.255.255 (all ones)

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example specifies an IP broadcast address of 0.0.0.0:

```
ip broadcast-address 0.0.0.0
```

## ip classless

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

**ip classless**  
**no ip classless**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

### Example

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

```
ip classless
```

## ip default-gateway

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

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

### Syntax Description

*ip-address* IP address of the router.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Cisco IOS software 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 back. The ICMP redirect message indicates which local router the Cisco IOS software should use.

### Example

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

```
ip default-gateway 192.31.7.18
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**show ip redirects**

## ip directed-broadcast

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

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

### Syntax Description

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

### Default

Enabled, with no list specified

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

### Example

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip forward-protocol**

## ip domain-list

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

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

### Syntax Description

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

### Default

No domain names are defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

### Examples

The following example adds several domain names to a list:

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

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip domain-name**

## ip domain-lookup

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

**ip domain-lookup**  
**no ip domain-lookup**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

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

```
ip domain-lookup
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip domain-lookup nsap**  
**ip domain-name**  
**ip name-server**

## ip domain-lookup nsap

To allow DNS queries for Connectionless Network System (CLNS) addresses, use the **ip domain-lookup nsap** global configuration command. To disable this feature, use the **no** form of this command.

**ip domain-lookup nsap**  
**no ip domain-lookup nsap**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

With both IP and International Organization for Standardization (ISO) CLNS enabled, this feature allows the Cisco IOS software to dynamically determine a CLNS address given a host name. This feature is useful for the ISO CLNS **ping EXEC** command and when making CLNS Telnet connections.

### Example

The following example disables DNS queries of CLNS addresses:

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip domain-lookup**  
**ping (for ISO CLNS)**

## ip domain-name

To define a default domain name that the Cisco IOS software uses to complete unqualified host names (names without a dotted-decimal domain name), use the **ip domain-name** global configuration command. To disable use of the DNS, use the **no** form of this command.

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

### Syntax Description

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

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

You can use the master indexes or search online to find documentation of related commands.

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

## ip forward-protocol

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

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

### Syntax Description

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

### Default

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

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

- Trivial File Transfer Protocol (TFTP) (port 69)
- Domain Naming System (port 53)
- Time service (port 37)
- NetBIOS Name Server (port 137)
- NetBIOS Datagram Server (port 138)
- Boot Protocol (BOOTP) client and server datagrams (ports 67 and 68)
- TACACS service (port 49)
- IEN-116 Name Service (port 42)

---

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

---

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

If you use the **ip forward-protocol** command, specifying just UDP, without the port, enables forwarding and flooding on the default ports.

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

### Example

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

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

## ip forward-protocol any-local-broadcast

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

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

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

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

### Example

Assume a router is directly connected to subnet 1 of network 131.108.0.0 and that the netmask is 255.255.255.0. The following command enables the forwarding of IP broadcasts destined to 131.108.1.255 and 131.108.1.0 in addition to the broadcast addresses mentioned in the “Usage Guidelines” section:

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip forward-protocol spanning-tree**

## ip forward-protocol spanning-tree

To permit IP broadcasts to be flooded throughout the internetwork in a controlled fashion, use the **ip forward-protocol spanning-tree** global configuration command. To disable the flooding of IP broadcasts, use the **no** form of this command.

```
ip forward-protocol spanning-tree  
no ip forward-protocol spanning-tree
```

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Packets must meet the following criteria to be considered for flooding:

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast; that is, an all-network broadcast (255.255.255.255) or major network broadcast (131.108.255.255, for example).
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP protocol specified by the **ip forward-protocol udp** global configuration command.
- The packet's time-to-live (TTL) value must be at least 2.

A flooded UDP datagram is given the destination address specified by the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any desired address. Thus, the destination address may change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

After a decision has been made to send the datagram out on an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is therefore subject to access lists, if they are present on the output interface.

The **ip forward-protocol spanning-tree** command uses the database created by the bridging spanning-tree protocol. Therefore, the transparent bridging option must be in the routing software, and bridging must be configured on each interface that is to participate in the flooding in order to support this capability.

If an interface does not have bridging configured, it still will be able to receive broadcasts, but it will never forward broadcasts received on that interface. Also, it will never use that interface to send broadcasts received on a different interface.

If no actual bridging is desired, you can configure a type-code bridging filter that will deny all packet types from being bridged. Refer to the “Configuring Transparent Bridging” chapter in the *Bridging and IBM Networking Configuration Guide* for more information about using access lists to filter bridged traffic. The spanning-tree database is still available to the IP forwarding code to use for the flooding.

The spanning-tree-based flooding mechanism forwards packets whose contents are all ones (255.255.255.255), all zeros (0.0.0.0), and, if subnetting is enabled, all networks (131.108.255.255 as an example in the network number 131.108.0.0). This mechanism also forward packets whose contents are the zeros version of the all-networks broadcast when subnetting is enabled (for example, 131.108.0.0).

This command is an extension of the **ip helper-address** interface configuration command, in that the same packets that may be subject to the helper address and forwarded to a single network can now be flooded. Only one copy of the packet will be put on each network segment.

### Example

The following example permits IP broadcasts to be flooded through the internetwork in a controlled fashion:

```
ip forward-protocol spanning-tree
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip broadcast-address**  
**ip forward-protocol**  
**ip forward-protocol turbo-flood**  
**ip helper-address**

## ip forward-protocol turbo-flood

To speed up flooding of User Datagram Protocol (UDP) datagrams using the spanning-tree algorithm, use the **ip forward-protocol turbo-flood** global configuration command. To disable this feature, use the **no** form of this command.

```
ip forward-protocol turbo-flood  
no ip forward-protocol turbo-flood
```

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Used in conjunction with the **ip forward-protocol spanning-tree** global configuration command, this feature is supported over Advanced Research Projects Agency (ARPA)-encapsulated Ethernets, FDDI, and HDLC-encapsulated serials, but is not supported on Token Rings. As long as the Token Rings and the non-HDLC serials are not part of the bridge group being used for UDP flooding, turbo flooding will behave normally.

### Example

The following is an example of a two-port router using this feature:

```
ip forward-protocol turbo-flood  
ip forward-protocol spanning-tree  
!  
interface ethernet 0  
  ip address 128.9.1.1  
  bridge-group 1  
!  
interface ethernet 1  
  ip address 128.9.1.2  
  bridge-group 1  
!  
bridge 1 protocol dec
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

```
ip forward-protocol  
ip forward-protocol spanning-tree
```

## ip helper-address

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

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

### Syntax Description

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

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

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

---

**Note** The **ip helper-address** command does not work on an X.25 interface on a destination router because the router cannot tell if the packet was intended as a physical broadcast.

---

### Example

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip forward-protocol**

## ip host

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

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

### Syntax Description

<i>name</i>	Name of the host. The first character can be either a letter or a number. If you use a number, the operations you can perform are limited.
<i>tcp-port-number</i>	(Optional) TCP port number to connect to when using the defined host name in conjunction with an EXEC connect or Telnet command. The default is Telnet (port 23).
<i>address1</i>	Associated IP address.
<i>address2...address8</i>	(Optional) Additional associated IP address. You can bind up to eight addresses to a host name.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

### Example

The following example defines 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

To configure your communication server to act as a terminal server, use the **ip host-routing** global configuration command. To disable host-based routing, use the **no** form of this command.

**ip host-routing**  
**no ip host-routing**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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 directly attached.

### 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

You can use the master indexes or search online to find documentation of related commands.

**ip host**  
**ip routing**

## ip hp-host

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

```
ip hp-host hostname ip-address  
no ip hp-host hostname ip-address
```

### Syntax Description

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

### Default

No host names are defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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 Commands

You can use the master indexes or search online to find documentation of related commands.

**ip probe proxy**

## ip irdp

To enable ICMP Router Discovery Protocol (IRDP) processing on an interface, use the **ip irdp** interface configuration command. To disable IRDP routing, use the **no** form of this command.

```
ip irdp [multicast | holdtime seconds | maxadvertinterval seconds | minadvertinterval seconds | preference number | address address [number]]  
no ip irdp
```

### Syntax Description

<b>multicast</b>	(Optional) Use the multicast address (224.0.0.1) instead of IP broadcasts.
<b>holdtime</b> <i>seconds</i>	(Optional) Length of time in seconds advertisements are held valid. Default is three times the <b>maxadvertinterval</b> value. Must be greater than <b>maxadvertinterval</b> and cannot be greater than 9000 seconds.
<b>maxadvertinterval</b> <i>seconds</i>	(Optional) Maximum interval in seconds between advertisements. The default is 600 seconds.
<b>minadvertinterval</b> <i>seconds</i>	(Optional) Minimum interval in seconds between advertisements. The default is 0.75 times the <b>maxadvertinterval</b> . If you change the <b>maxadvertinterval</b> value, this value defaults to three-quarters of the new value.
<b>preference</b> <i>number</i>	(Optional) Preference value. The allowed range is $-2^{31}$ to $2^{31}$ . The default is 0. A higher value increases the router's preference level. You can modify a particular router so that it will be the preferred router to which others home.
<b>address</b> <i>address</i> [ <i>number</i> ]	(Optional) IP address ( <i>address</i> ) to proxy-advertise, and optionally, its preference value ( <i>number</i> ).

### Default

Disabled

When enabled, IRDP uses these defaults:

- Broadcast IRDP advertisements
- Maximum interval between advertisements: 600 seconds
- Minimum interval between advertisements: 0.75 times **maxadvertinterval**
- Preference: 0

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

If you change **maxadvertinterval**, the other two values also change, so it is important to change **maxadvertinterval** first before changing either **holdtime** or **minadvertinterval**.

The **ip irdp multicast** command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.

## Example

The following example sets the various IRDP processes:

```
! enable irdp on interface Ethernet 0
interface ethernet 0
  ip irdp
  ! send IRDP advertisements to the multicast address
  ip irdp multicast
  ! increase router preference from 100 to 50
  ip irdp preference 50
  ! set maximum time between advertisements to 400 secs
  ip irdp maxadvertinterval 400
  ! set minimum time between advertisements to 100 secs
  ip irdp minadvertinterval 100
  ! advertisements are good for 6000 seconds
  ip irdp holdtime 6000
  ! proxy-advertise 131.108.14.5 with default router preference
  ip irdp address 131.108.14.5
  ! proxy-advertise 131.108.14.6 with preference of 50
  ip irdp address 131.108.14.6 50
```

## Related Commands

You can use the master indexes or search online to find documentation of related commands.

### **show ip irdp**

## ip mobile arp

To enable local-area mobility, use the **ip mobile arp** interface configuration command. To disable local-area mobility, use the **no** form of this command.

**ip mobile arp** [**timers** *keepalive hold-time*] [**access-group** *access-list-number* | *name*]  
**no ip mobile arp** [**timers** *keepalive hold-time*] [**access-group** *access-list-number* | *name*]

### Syntax Description

<b>timers</b>	(Optional) Indicates that you are setting local-area mobility timers.
<i>keepalive</i>	(Optional) Frequency, in seconds, at which the Cisco IOS software sends unicast ARP messages to a relocated host to verify that the host is present and has not moved. The default keepalive time is 300 seconds (5 minutes).
<i>hold-time</i>	(Optional) Hold time, in seconds. This is the length of time the software considers that a relocated host is present without receiving some type of ARP broadcast or unicast from the host. Normally, the hold time should be at least three times greater than the keepalive time. The default hold time is 900 seconds (15 minutes).
<b>access-group</b>	(Optional) Indicates that you are applying an access list. This access list applies only to local-area mobility.
<i>access-list-number</i>	(Optional) Number of a standard IP access list. It is a decimal number from 1 to 99. Only hosts with addresses permitted by this access list are accepted for local-area mobility.
<i>name</i>	(Optional) Name of an IP access list. The name cannot contain a space or quotation mark, and must begin with an alphabetic character to avoid ambiguity with numbered access lists.

### Defaults

Local-area mobility is disabled.

If you enable local-area mobility:

*keepalive*: 300 seconds (5 minutes)

*hold-time*: 900 seconds (15 minutes)

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

Local-area mobility is supported on Ethernet, Token Ring, and FDDI interfaces only.

To create larger mobility areas, you must first redistribute the mobile routes into your Interior Gateway Protocol (IGP). The IGP must support host routes. You can use Enhanced IGRP, OSPF, or Intermediate System-to-Intermediate System (IS-IS); you can also use RIP, but this is not recommended. The mobile area must consist of a contiguous set of subnets.

Using an access list to control the list of possible mobile nodes is strongly encouraged. Without an access list, misconfigured hosts can be taken for mobile nodes and disrupt normal operations.

## Example

The following example configures local-area mobility on Ethernet interface 0:

```
bridge 1 protocol ieee
access-list 10 permit 198.92.37.114
interface ethernet 0
 ip mobile arp access-group 10
 bridge-group 1
```

## Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (standard)**

**bridge-group**

**bridge protocol**

**default-metric (BGP, EGP, OSPF, and RIP)**

**network (BGP)**

**network (EGP)**

**network (IGRP)**

**network (RIP)**

**redistribute**

**router eigrp**

**router isis**

**router ospf**

## ip name-server

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

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

### Syntax Description

<i>server-address1</i>	IP addresses of name server.
<i>server-address2...server-address6</i>	(Optional) IP addresses of additional name servers (a maximum of six name servers).

### Default

No name server addresses are specified.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0

### Example

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

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

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip domain-lookup**  
**ip domain-name**

## ip nat

To designate that traffic originating from or destined for the interface is subject to Network Address Translation (NAT), use the **ip nat** interface configuration command. To prevent the interface from being able to translate, use the **no** form of this command.

```
ip nat { inside | outside }  
no ip nat { inside | outside }
```

### Syntax Description

**inside** Indicates the interface is connected to the inside network (the network subject to NAT translation).

**outside** Indicates the interface is connected to the outside network.

### Default

Traffic leaving or arriving at this interface is not subject to network address translation.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Only packets moving between “inside” and “outside” interfaces can be translated. You must specify at least one inside interface and outside interface for each border router where you intend to use NAT.

### Example

The following example translates between inside hosts addressed from either the 192.168.1.0 or 192.168.2.0 networks to the globally unique 171.69.233.208/28 network:

```
ip nat pool net-208 171.69.233.208 171.69.233.223 prefix-length 28  
ip nat inside source list 1 pool net-208  
!  
interface ethernet 0  
 ip address 171.69.232.182 255.255.255.240  
 ip nat outside  
!  
interface ethernet 1  
 ip address 192.168.1.94 255.255.255.0  
 ip nat inside  
!  
access-list 1 permit 192.168.1.0 0.0.0.255  
access-list 1 permit 192.168.2.0 0.0.0.255
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**  
**ip nat inside destination**  
**ip nat inside source**  
**ip nat outside source**  
**ip nat pool**  
**ip nat translation**  
**show ip nat statistics**  
**show ip nat translations**

## ip nat inside destination

To enable Network Address Translation (NAT) of the inside destination address, use the **ip nat inside destination** global configuration command. To remove the dynamic association to a pool, use the **no ip nat inside destination** form of this command.

```
ip nat inside destination list {access-list-number | name} pool name
no ip nat inside destination list {access-list-number | name}
```

### Syntax Description

<b>list</b> <i>access-list-number</i>	Standard IP access list number. Packets with destination addresses that pass the access list are translated using global addresses from the named pool.
<b>list</b> <i>name</i>	Name of a standard IP access list. Packets with destination addresses that pass the access list are translated using global addresses from the named pool.
<b>pool</b> <i>name</i>	Name of the pool from which global IP addresses are allocated during dynamic translation.

### Default

No inside destination addresses are translated.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

This command has two forms: dynamic and static address translation. The form with an access list establishes dynamic translation. Packets from addresses that match the standard access list are translated using global addresses allocated from the pool named with the **ip nat pool** command.

### Example

The following example translates between inside hosts addressed to either the 192.168.1.0 or 192.168.2.0 networks to the globally unique 171.69.233.208/28 network:

```
ip nat pool net-208 171.69.233.208 171.69.233.223 prefix-length 28
ip nat inside destination list 1 pool net-208
!
interface ethernet 0
 ip address 171.69.232.182 255.255.255.240
 ip nat outside
!
interface ethernet 1
 ip address 192.168.1.94 255.255.255.0
 ip nat inside
!
access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**

**ip nat**

**ip nat inside source**

**ip nat outside source**

**ip nat pool**

**ip nat translation**

**show ip nat statistics**

**show ip nat translations**

## ip nat inside source

To enable Network Address Translation (NAT) of the inside source address, use the **ip nat inside source** global configuration command. To remove the static translation or remove the dynamic association to a pool, use the **no** form of this command.

```
ip nat inside source {list {access-list-number | name} pool name [overload] | static local-ip
global-ip}
no ip nat inside source {list {access-list-number | name} pool name [overload] | static local-ip
global-ip}
```

### Syntax Description

<b>list</b> <i>access-list-number</i>	Standard IP access list number. Packets with source addresses that pass the access list are dynamically translated using global addresses from the named pool.
<b>list</b> <i>name</i>	Name of a standard IP access list. Packets with source addresses that pass the access list are dynamically translated using global addresses from the named pool.
<b>pool</b> <i>name</i>	Name of the pool from which global IP addresses are allocated dynamically.
<b>overload</b>	(Optional) Enables the router to use one global address for many local addresses. When overloading is configured, each inside host's TCP or UDP port number distinguishes between the multiple conversations using the same local IP address.
<b>static</b> <i>local-ip</i>	Sets up a single static translation; this argument establishes the local IP address assigned to a host on the inside network. The address could be randomly chosen, allocated from RFC 1918, or obsolete.
<i>global-ip</i>	Sets up a single static translation; this argument establishes the globally unique IP address of an inside host as it appears to the outside world.

### Default

No NAT translation of inside source addresses occurs.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

This command has two forms: dynamic and static address translation. The form with an access list establishes dynamic translation. Packets from addresses that match the standard access list are translated using global addresses allocated from the pool named with the **ip nat pool** command.

Alternatively, the syntax form with the keyword **static** establishes a single static translation.

### Example

The following example translates between inside hosts addressed from either the 192.168.1.0 or 192.168.2.0 networks to the globally unique 171.69.233.208/28 network:

```
ip nat pool net-208 171.69.233.208 171.69.233.223 prefix-length 28
ip nat inside source list 1 pool net-208
!
interface ethernet 0
 ip address 171.69.232.182 255.255.255.240
 ip nat outside
!
interface ethernet 1
 ip address 192.168.1.94 255.255.255.0
 ip nat inside
!
access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**

**ip nat**

**ip nat inside destination**

**ip nat outside source**

**ip nat pool**

**ip nat translation**

**show ip nat statistics**

**show ip nat translations**

## ip nat outside source

To enable Network Address Translation (NAT) of the outside source address, use the **ip nat outside source** global configuration command. To remove the static entry or the dynamic association, use the **no** form of this command.

```
ip nat outside source {list {access-list-number | name} pool name | static global-ip local-ip}
no ip nat outside source {list {access-list-number | name} pool name | static global-ip local-ip}
```

### Syntax Description

<b>list</b> <i>access-list-number</i>	Standard IP access list number. Packets with source addresses that pass the access list are translated using global addresses from the named pool.
<b>list</b> <i>name</i>	Name of a standard IP access list. Packets with source addresses that pass the access list are translated using global addresses from the named pool.
<b>pool</b> <i>name</i>	Name of the pool from which global IP addresses are allocated.
<b>static</b> <i>global-ip</i>	Sets up a single static translation. This argument establishes the globally unique IP address assigned to a host on the outside network by its owner. It was allocated from globally routable network space.
<i>local-ip</i>	Sets up a single static translation. This argument establishes the local IP address of an outside host as it appears to the inside world. The address was allocated from address space routable on the inside (RFC 1918, perhaps).

### Default

No translation of source addresses coming from the outside to the inside network occurs.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

You might have IP addresses that are not legal, officially assigned IP addresses. Perhaps you chose IP addresses that officially belong to another network. The case of an address used illegally and legally is called *overlapping*. You can use NAT to translate inside addresses that overlap with outside addresses. Use this feature if your IP addresses in the stub network happen to be legitimate IP addresses belonging to another network, and you need to communicate with those hosts or routers.

This command has two forms: dynamic and static address translation. The form with an access list establishes dynamic translation. Packets from addresses that match the standard access list are translated using global addresses allocated from the pool named with the **ip nat pool** command.

Alternatively, the syntax form with the keyword **static** establishes a single static translation.

### Example

The following example translates between inside hosts addressed from the 9.114.11.0 network to the globally unique 171.69.233.208/28 network. Further packets from outside hosts addressed from the 9.114.11.0 network (the true 9.114.11.0 network) are translated to appear to be from the network 10.0.1.0/24.

```
ip nat pool net-208 171.69.233.208 171.69.233.223 prefix-length 28
ip nat pool net-10 10.0.1.0 10.0.1.255 prefix-length 24
ip nat inside source list 1 pool net-208
ip nat outside source list 1 pool net-10
!
interface ethernet 0
 ip address 171.69.232.182 255.255.255.240
 ip nat outside
!
interface ethernet 1
 ip address 9.114.11.39 255.255.255.0
 ip nat inside
!
access-list 1 permit 9.114.11.0 0.0.0.255
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**

**ip nat**

**ip nat inside destination**

**ip nat inside source**

**ip nat pool**

**ip nat translation**

**show ip nat statistics**

**show ip nat translations**

## ip nat pool

To define a pool of IP addresses for Network Address Translation (NAT), use the **ip nat pool** global configuration command. To remove one or more addresses from the pool, use the **no** form of this command.

```
ip nat pool name start-ip end-ip {netmask netmask | prefix-length prefix-length}
[type rotary]
no ip nat pool name start-ip end-ip {netmask netmask | prefix-length prefix-length}
[type rotary]
```

### Syntax Description

<i>name</i>	Name of the pool.
<i>start-ip</i>	Starting IP address that defines the range of addresses in the address pool.
<i>end-ip</i>	Ending IP address that defines the range of addresses in the address pool.
<b>netmask</b> <i>netmask</i>	Network mask that indicates which address bits belong to the network and subnetwork fields and which bits belong to the host field. Specify the netmask of the network to which the pool addresses belong.
<b>prefix-length</b> <i>prefix-length</i>	Number that indicates how many bits of the netmask are ones (how many bits of the address indicate network). Specify the netmask of the network to which the pool addresses belong.
<b>type rotary</b>	(Optional) Indicates that the range of address in the address pool identify real, inside hosts among which TCP load distribution will occur.

### Default

No pool of addresses is defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

This command defines a pool of addresses using start address, end address, and either netmask or prefix length. The pool could define either an inside global pool, an outside local pool, or a rotary pool.

### Example

The following example translates between inside hosts addressed from either the 192.168.1.0 or 192.168.2.0 networks to the globally unique 171.69.233.208/28 network:

```
ip nat pool net-208 171.69.233.208 171.69.233.223 prefix-length 28
ip nat inside source list 1 pool net-208
!
interface ethernet 0
```

```
ip address 171.69.232.182 255.255.255.240
ip nat outside
!
interface ethernet 1
ip address 192.168.1.94 255.255.255.0
ip nat inside
!
access-list 1 permit 192.168.1.0 0.0.0.255
access-list 1 permit 192.168.2.0 0.0.0.255
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**

**ip nat**

**ip nat inside destination**

**ip nat inside source**

**ip nat outside source**

**ip nat translation**

**show ip nat statistics**

**show ip nat translations**

## ip nat translation

To change the amount of time after which Network Address Translation (NAT) translations time out, use the **ip nat translation** global configuration command. To disable the timeout, use the **no** form of this command.

```
ip nat translation { timeout | udp-timeout | dns-timeout | tcp-timeout | finrst-timeout }
    seconds
no ip nat translation { timeout | udp-timeout | dns-timeout | tcp-timeout | finrst-timeout }
```

### Syntax Description

<b>timeout</b>	Specifies that the timeout value applies to dynamic translations except for overload translations. Default is 86400 seconds (24 hours).
<b>udp-timeout</b>	Specifies that the timeout value applies to the UDP port. Default is 300 seconds (5 minutes).
<b>dns-timeout</b>	Specifies that the timeout value applies to connections to the Domain Naming System (DNS). Default is 60 seconds.
<b>tcp-timeout</b>	Specifies that the timeout value applies to the TCP port. Default is 86400 seconds (24 hours).
<b>finrst-timeout</b>	Specifies that the timeout value applies to Finish and Reset TCP packets, which terminate a connection. Default is 60 seconds.
<i>seconds</i>	Number of seconds after which the specified port translation times out. Default values are listed in the Default section.

### Defaults

**timeout** is 86400 seconds (24 hours)  
**udp-timeout** is 300 seconds (5 minutes)  
**dns-timeout** is 60 seconds (1 minute)  
**tcp-timeout** is 86400 seconds (24 hours)  
**finrst-timeout** is 60 seconds (1 minute)

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

When port translation is configured, there is finer control over translation entry timeouts because each entry contains more context about the traffic that is using it. Non-Domain Naming System UDP translations time out after 5 minutes, while DNS times out in 1 minute. TCP translations timeout in 24 hours, unless a RST or FIN is seen on the stream, in which case they will time out in 1 minute.

### Example

The following example causes UDP port translation entries to timeout after 10 minutes:

```
ip nat translation udp-timeout 600
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**

**ip nat**

**ip nat inside destination**

**ip nat inside source**

**ip nat outside source**

**ip nat pool**

**show ip nat statistics**

**show ip nat translations**

## ip netmask-format

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

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

### Syntax Description

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

### Default

Netmasks are displayed in bitcount format.

### Command Mode

Line configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

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

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

### Example

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

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

## ip nhrp authentication

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

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

### Syntax Description

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

### Default

No authentication string is configured; the Cisco IOS software adds no authentication option to NHRP packets it generates.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

All routers configured with NHRP within one logical NBMA network must share the same authentication string.

### Example

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

```
ip nhrp authentication specialxx
```

## ip nhrp holdtime

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

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

### Syntax Description

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

### Default

7200 seconds (2 hours) for both arguments

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

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

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

### Examples

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

```
ip nhrp holdtime 3600
```

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

```
ip nhrp holdtime 7200 3600
```

## ip nhrp interest

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

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

### Syntax Description

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

### Default

All non-NHRP packets can trigger NHRP requests.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

The **ip nhrp interest** command controls *which* packets cause NHRP address resolution to take place; the **ip nhrp use** command controls *how readily* the system attempts such address resolution.

### Example

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**  
**access-list (standard)**  
**ip nhrp use**

## ip nhrp map

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

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

### Syntax Description

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

### Default

No static IP-to-NBMA cache entries exist.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

### Example

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

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

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nhrp**

## ip nhrp map multicast

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

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

### Syntax Description

*nbma-address* Nonbroadcast, multiaccess (NBMA) address which is directly reachable through the NBMA network. The address format varies depending on the medium you are using.

### Default

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

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3, and applies only to tunnel interfaces.

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

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

### Example

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

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

## ip nhrp max-send

To change the maximum frequency at which NHRP packets can be sent, use the **ip nhrp max-send** interface configuration command. To restore this frequency to the default value, use the **no** form of this command.

```
ip nhrp max-send pkt-count every interval  
no ip nhrp max-send
```

### Syntax Description

<i>pkt-count</i>	Number of packets which can be transmitted in the range from 1 to 65535. Default is 5 packets.
<b>every</b> <i>interval</i>	Time (in seconds) in the range from 10 to 65535. Default is 10 seconds.

### Defaults

*pkt-count* = 5 packets  
*interval* = 10 seconds

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.1.

The software maintains a per-interface quota of NHRP packets that can be transmitted. NHRP traffic, whether locally generated or forwarded, cannot be sent at a rate that exceeds this quota. The quota is replenished at the rate specified by *interval*.

### Example

In the following example, only 1 NHRP packet can be sent from serial interface 0 each minute:

```
interface serial 0  
  ip nhrp max-send 1 every 60
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip nhrp interest**  
**ip nhrp use**

## ip nhrp network-id

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

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

### Syntax Description

*number* Globally unique, 32-bit network identifier for a nonbroadcast, multiaccess (NBMA) network. The range is 1 to 4294967295.

### Default

NHRP is disabled on the interface.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

In general, all NHRP stations within one logical NBMA network must be configured with the same network identifier.

### Example

The following example enables NHRP on the interface:

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

## ip nhrp nhs

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

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

### Syntax Description

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

### Default

No Next Hop Servers are explicitly configured, so normal network layer routing decisions are used to forward NHRP traffic.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

Use this command to specify the address of a Next Hop Server and the networks it serves. Normally, NHRP consults the network layer forwarding table to determine how to forward NHRP packets. When Next Hop Servers are configured, these next hop addresses override the forwarding path that would otherwise be used for NHRP traffic.

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

### Example

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

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

## ip nhrp record

To re-enable the use of forward record and reverse record options in NHRP Request and Reply packets, use the **ip nhrp record** interface configuration command. To suppress the use of such options, use the **no** form of this command.

**ip nhrp record**  
**no ip nhrp record**

### Syntax Description

This command has no arguments or keywords.

### Default

Forward record and reverse record options are used in NHRP Request and Reply packets.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

### Example

The following example suppresses forward record and reverse record options:

```
no ip nhrp record
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip nhrp responder**

## ip nhrp responder

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

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

### Syntax Description

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

### Default

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

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

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

### Example

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

```
ip nhrp responder serial 0
```

## ip nhrp use

To configure the software so that NHRP is deferred until the system has attempted to send data traffic to a particular destination multiple times, use the **ip nhrp use** interface configuration command. To restore the default value, use the **no** form of this command.

```
ip nhrp use usage-count  
no ip nhrp use usage-count
```

### Syntax Description

*usage-count* Packet count in the range from 1 to 65535. Default is 1.

### Default

*usage-count* = 1. The first time a data packet is sent to a destination for which the system determines NHRP can be used, an NHRP request is sent.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.1.

When the software attempts to transmit a data packet to a destination for which it has determined that NHRP address resolution can be used, an NHRP request for that destination is normally transmitted right away. Configuring the *usage-count* causes the system to wait until that many data packets have been sent to a particular destination before it attempts NHRP. The *usage-count* for a particular destination is measured over 1-minute intervals (the NHRP cache expiration interval).

The usage-count applies *per destination*. So if *usage-count* is configured to be 3, and 4 data packets are sent toward 10.0.0.1 and 1 packet toward 10.0.0.2, then an NHRP request is generated for 10.0.0.1 only.

If the system continues to need to forward data packets to a particular destination, but no NHRP response has been received, retransmission of NHRP requests are performed. This retransmission occurs only if data traffic continues to be sent to a destination.

The **ip nhrp interest** command controls *which* packets cause NHRP address resolution to take place; the **ip nhrp use** command controls *how readily* the system attempts such address resolution.

### Example

In the following example, if in the first minute 4 packets are sent to one destination and 5 packets are sent to a second destination, then a single NHRP request is generated for the second destination.

If in the second minute the same traffic is generated and no NHRP responses have been received, then the system retransmits its request for the second destination.

```
ip nhrp use 5
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip nhrp interest**

**ip nhrp max-send**

## ip probe proxy

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

**ip probe proxy**  
**no ip probe proxy**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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 Commands

You can use the master indexes or search online to find documentation of related commands.

**ip hp-host**

## ip proxy-arp

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

**ip proxy-arp**  
**no ip proxy-arp**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example enables proxy ARP on Ethernet interface 0:

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

## ip redirects

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

**ip redirects**  
**no ip redirects**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled, unless Hot Standby Router Protocol is configured

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

If the Hot Standby Router Protocol is configured on an interface, ICMP Redirect messages are disabled by default for the interface.

### Example

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

```
interface ethernet 0
 ip redirects
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**show ip redirects**

## ip routing

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

**ip routing**  
**no ip routing**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

If CRB is configured, this command is unnecessary because all protocols are bridged by default.

If CRB is not configured, configure the **no ip routing** command to bridge IP.

### Example

The following example enables IP routing:

```
ip routing
```

## ip subnet-zero

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

**ip subnet-zero**  
**no ip subnet-zero**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

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

### Example

In the following example, subnet-zero is enabled:

```
ip subnet-zero
```

## ip unnumbered

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

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

### Syntax Description

*type number* Type and number of another interface on which the router has an assigned IP address. It cannot be another unnumbered interface.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

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

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

---

**Note** Using an unnumbered serial line between different major networks (or *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
```

## ping (privileged)

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

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

### Syntax Description

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

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

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

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

```
ping 128.111.3.255
```

As a side effect, you also can get a list of all multicast-capable hosts that are connected directly to the router 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 describes the test characters that the ping facility sends.

**Table 1 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.

**Table 1 Ping Test Characters (Continued)**

Char	Description
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.

```
Router# 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 2 describes significant fields shown in the display.

**Table 2 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 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.

**Table 2 IP Ping Internet Header Options Field Descriptions (Continued)**

Field	Description
Set DF bit in IP header?	Don't Fragment. Specifies that if the packet encounters a node in its path that is configured for a smaller MTU than the packet's MTU, that the packet is to be dropped and an error message is to be sent to the router at the packet's source address. If performance problems are encountered on the network, a node configured for a small MTU could be a contributing factor. This feature can be used to determine the smallest MTU in the path. Default: no.
Data pattern [0xABCD]:	Sets 16-bit hexadecimal data pattern. Default: 0xABCD. Varying the data pattern in this field (to all ones or all zeros for example) can be useful when debugging data sensitivity problems on CSU/DSUs, or detecting cable-related problems such as cross talk.
Loose, Strict, Record, Timestamp, Verbose [none]:	Supported Internet header options. The Cisco IOS software examines the header options to every packet that passes through it. If it finds a packet with an invalid option, the software sends an ICMP Parameter Problem message to the source of the packet and discards the packet. The Internet header options are as follows: <ul style="list-style-type: none"> <li>• Loose</li> <li>• Strict</li> <li>• Record (see the following section for more information on this helpful option)</li> <li>• Timestamp</li> <li>• Verbose</li> </ul> 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 <b>ping</b> output display, depending on the protocol type.
Success rate is 100 percent	Percentage of packets successfully echoed back to the router. 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:

```
Router# ping

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

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

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

1 in 8 ms. Received packet has options
Total option bytes= 4 padded length=40
Record route: 160.89.80.31 131.108.6.10 131.108.1.6 131.108.1.115
              131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
End of list

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

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

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

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

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 to 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, where 0 is the number assigned to this packet to indicate that it is the first in the series, and 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 router 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 Commands

You can use the master indexes or search online to find documentation of related commands.

## ping (user)

## ping (user)

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

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

### Syntax Description

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

### Command Mode

User EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

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

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

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

In the **ping (privileged)** section, Table 1 describes the test characters that the ping facility sends.

### Sample Display Using an IP Host Name

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

```
Router> 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:

```
Router> 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 Commands

You can use the master indexes or search online to find documentation of related commands.

### ping (privileged)

## show arp

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

```
show arp
```

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

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

```
Router# show arp

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

Table 3 describes significant fields shown in the first line of output in the display.

**Table 3 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 Cisco IOS software is using for the network address in this entry. Possible values include: <ul style="list-style-type: none"> <li>• ARPA</li> <li>• SNAP</li> <li>• ETLK (EtherTalk)</li> <li>• SMDS</li> </ul>
Interface	Indicates the interface associated with this network address.

## show hosts

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

### show hosts

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

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

```
Router# show hosts

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

Table 4 describes significant fields shown in the display.

**Table 4 Show Hosts Field Descriptions**

Field	Description
Flag	A temporary entry is entered by a name server; the Cisco IOS software removes the entry after 72 hours of inactivity. A permanent 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 software last referred to the cache entry.
Type	Identifies the type of address, for example, IP, CLNS, or X.121. If you have used the <b>ip hp-host</b> global configuration command, the <b>show hosts</b> 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 Commands

You can use the master indexes or search online to find documentation of related commands.

**clear host**

## show ip aliases

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

```
show ip aliases
```

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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:

```
Router# show ip aliases

      IP Address      Port
131.108.29.245      SLIP TTY1
```

The display lists the IP address and corresponding port number.

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**show line**

## show ip arp

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

```
show ip arp [ip-address] [hostname] [mac-address] [type number]
```

### Syntax Description

<i>ip-address</i>	(Optional) ARP entries matching this IP address are displayed.
<i>hostname</i>	(Optional) Host name.
<i>mac-address</i>	(Optional) 48-bit MAC address.
<i>type number</i>	(Optional) ARP entries learned via this interface type and number are displayed.

### 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:

```
Router# show ip arp

Protocol  Address           Age (min)  Hardware Addr  Type   Interface
-----  -
Internet  171.69.233.22     9          0000.0c59.f892  ARPA   Ethernet0/0
Internet  171.69.233.21     8          0000.0c07.ac00  ARPA   Ethernet0/0
Internet  171.69.233.19     -          0000.0c63.1300  ARPA   Ethernet0/0
Internet  171.69.233.30     9          0000.0c36.6965  ARPA   Ethernet0/0
Internet  172.19.168.11     -          0000.0c63.1300  ARPA   Ethernet0/0
Internet  172.19.168.254   9          0000.0c36.6965  ARPA   Ethernet0/0
```

Table 5 describes significant fields shown in the display.

**Table 5 Show IP ARP Field Descriptions**

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.

**Table 5 Show IP ARP Field Descriptions (Continued)**

<b>Field</b>	<b>Description</b>
Type	Type of encapsulation: <ul style="list-style-type: none"><li>• ARPA—Ethernet</li><li>• SNAP—RFC 1042</li><li>• SAP—IEEE 802.3</li></ul>
Interface	Interface to which this address mapping has been assigned.

## show ip interface

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

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

### Syntax Description

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

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Cisco IOS software automatically enters a directly connected route in the routing table if the interface is usable. A usable interface is one through which the software can send and receive packets. If the software determines that an interface is not usable, it removes the directly connected routing entry from the routing table. Removing the entry allows the software 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 arguments, you will see information on all the interfaces.

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

### Sample Display

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

```
Router# show ip interface

Ethernet0 is up, line protocol is up
  Internet address is 192.195.78.24, subnet mask is 255.255.255.240
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Secondary address 131.192.115.2, subnet mask 255.255.255.0
  Directed broadcast forwarding is enabled
  Multicast groups joined: 224.0.0.1 224.0.0.2
  Outgoing access list is not set
  Inbound access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
```

```

ICMP unreachable are always sent
ICMP mask replies are never sent
IP fast switching is enabled
IP fast switching on the same interface is disabled
IP SSE switching is disabled
Router Discovery is disabled
IP output packet accounting is disabled
IP access violation accounting is disabled
TCP/IP header compression is disabled
Probe proxy name replies are disabled

```

Table 6 describes the fields shown in the display.

**Table 6 Show IP Interface Field Descriptions**

Field	Description
Ethernet0 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.
Internet address and subnet mask	IP Internet address and subnet mask of the interface.
Broadcast address	Shows the broadcast address.
Address determined by ...	Indicates how the IP address of the interface was determined.
MTU	Shows the MTU value set on the interface.
Helper address	Shows a helper address, if one has been set.
Secondary address	Shows a secondary address, if one has been set.
Directed broadcast forwarding	Indicates whether directed broadcast forwarding is enabled.
Multicast groups joined	Indicates the multicast groups this interface is a member of.
Outgoing access list	Indicates whether the interface has an outgoing access list set.
Inbound access list	Indicates whether the interface has an incoming 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.
Split horizon	Indicates split horizon is enabled.
ICMP redirects	Specifies whether redirects will be sent on this interface.
ICMP unreachable	Specifies whether unreachable messages will be sent on this interface.
ICMP mask replies	Specifies whether mask replies will be sent on this interface.
IP fast switching	Specifies whether fast switching has been enabled for this interface. It is generally enabled on serial interfaces, such as this one.
IP SSE switching	Specifies whether IP SSE switching is enabled.
Router Discovery	Specifies whether the discovery process has been enabled for this interface. It is generally disabled on serial interfaces.
IP output packet 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 irdp

To display IRDP values, use the **show ip irdp** EXEC command.

```
show ip irdp
```

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

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

```
Router# show ip irdp

Ethernet 0 has router discovery enabled

Advertisements will occur between every 450 and 600 seconds.
Advertisements are valid for 1800 seconds.
Default preference will be 100.
--More--
Serial 0 has router discovery disabled
--More--
Ethernet 1 has router discovery disabled
```

As the display shows, **show ip irdp** output indicates whether router discovery has been configured for each router interface, and it lists the values of router discovery configurables for those interfaces on which router discovery has been enabled. Explanations for the less self-evident lines of output in the display are as follows:

```
Advertisements will occur between every 450 and 600 seconds.
```

This indicates the configured minimum and maximum advertising interval for the interface.

```
Advertisements are valid for 1800 seconds.
```

This indicates the configured holdtime values for the interface.

```
Default preference will be 100.
```

This indicates the configured (or in this case default) preference value for the interface.

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip irdp**

## show ip masks

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

**show ip masks** *address*

### Syntax Description

*address* Network address for which a mask is required.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The **show ip masks** command is useful for debugging when a variable-length subnet mask (VLSM) is 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:

```
Router# 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 nat statistics

To display Network Address Translation (NAT) statistics, use the **show ip nat statistics EXEC** command.

**show ip nat statistics**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Sample Display

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

```
Router# show ip nat statistics
Total translations: 2 (0 static, 2 dynamic; 0 extended)
Outside interfaces: Serial0
Inside interfaces: Ethernet1
Hits: 135 Misses: 5
Expired translations: 2
Dynamic mappings:
-- Inside Source
access-list 1 pool net-208 refcount 2
pool net-208: netmask 255.255.255.240
start 171.69.233.208 end 171.69.233.221
type generic, total addresses 14, allocated 2 (14%), misses 0
```

Table 7 describes the significant fields in the display.

**Table 7 Show IP NAT Statistics Field Descriptions**

Field	Description
Total translations	Number of translations active in the system. This number is incremented each time a translation is created and is decremented each time a translation is cleared or times out.
Outside interfaces	List of interfaces marked as outside with the <b>ip nat outside</b> command.
Inside interfaces	List of interfaces marked as inside with the <b>ip nat inside</b> command.
Hits	Number of times the software does a translations table lookup and finds an entry.
Misses	Number of times the software does a translations table lookup, fails to find an entry, and must try to create one.
Expired translations	Cumulative count of translations that have expired since the router was booted.
Dynamic mappings	Indicates that the information that follows is about dynamic mappings.

**Table 7 Show IP NAT Statistics Field Descriptions (Continued)**

<b>Field</b>	<b>Description</b>
Inside Source	The information that follows is about an inside source translation.
access-list	Access list number being used for the translation.
pool	Name of the pool (in this case, net-208).
refcount	Number of translations that are using this pool.
netmask	IP network mask being used in the pool.
start	Starting IP address in the pool range.
end	Ending IP address in the pool range.
type	Type of pool. Possible types are generic or rotary.
total addresses	Number of addresses in the pool that are available for translation.
allocated	Number of addresses being used.
misses	Number of failed allocations from the pool.

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**

**ip nat**

**ip nat inside destination**

**ip nat inside source**

**ip nat outside source**

**ip nat pool**

**ip nat translation**

**show ip nat statistics**

**show ip nat translations**

## show ip nat translations

To display active Network Address Translation (NAT) translations, use the **show ip nat translations EXEC** command.

**show ip nat translations [verbose]**

### Syntax Description

**verbose** (Optional) Displays additional information for each translation table entry, including how long ago the entry was created and used.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Sample Displays

The following is sample output from the **show ip nat translations** command. Without overloading, two inside hosts are exchanging packets with some number of outside hosts.

```
Router# show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
--- 171.69.233.209     192.168.1.95     ---                ---
--- 171.69.233.210     192.168.1.89     ---                --
```

With overloading, a translation for a DNS transaction is still active, and translations for two Telnet sessions (from two different hosts) are also active. Note that two different inside hosts appear on the outside with a single IP address.

```
Router# show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
udp 171.69.233.209:1220 192.168.1.95:1220 171.69.2.132:53    171.69.2.132:53
tcp 171.69.233.209:11012 192.168.1.89:11012 171.69.1.220:23    171.69.1.220:23
tcp 171.69.233.209:1067 192.168.1.95:1067 171.69.1.161:23    171.69.1.161:23
```

The following is sample output that includes the **verbose** keyword.

```
Router# show ip nat translations verbose
Pro Inside global      Inside local      Outside local      Outside global
udp 171.69.233.209:1220 192.168.1.95:1220 171.69.2.132:53    171.69.2.132:53
    create 00:00:02, use 00:00:00, flags: extended
tcp 171.69.233.209:11012 192.168.1.89:11012 171.69.1.220:23    171.69.1.220:23
    create 00:01:13, use 00:00:50, flags: extended
tcp 171.69.233.209:1067 192.168.1.95:1067 171.69.1.161:23    171.69.1.161:23
    create 00:00:02, use 00:00:00, flags: extended
```

Table 8 describes the significant fields in the display.

**Table 8 Show IP NAT Translations Field Descriptions**

Field	Description
Pro	Protocol of the port identifying the address.
Inside global	The legitimate IP address (assigned by the NIC or service provider) that represents one or more inside local IP addresses to the outside world.
Inside local	The IP address assigned to a host on the inside network; probably not a legitimate address assigned by the NIC or service provider.
Outside local	IP address of an outside host as it appears to the inside network; probably not a legitimate address assigned by the NIC or service provider.
Outside global	The IP address assigned to a host on the outside network by its owner.
create	How long ago the entry was created (in hours:minutes:seconds).
use	How long ago the entry was last used (in hours:minutes:seconds).
flags	Indication of the type of translation. Possible flags are <ul style="list-style-type: none"><li>• extended—Extended translation</li><li>• static—Static translation</li><li>• destination—Rotary translation</li><li>• outside—Outside translation</li><li>• timing out—Translation will no longer be used, due to a TCP FIN or RST.</li></ul>

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear ip nat translation**  
**ip nat**  
**ip nat inside destination**  
**ip nat inside source**  
**ip nat outside source**  
**ip nat pool**  
**ip nat translation**  
**show ip nat statistics**

## show ip nhrp

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

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

### Syntax Description

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

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Sample Display

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

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

Table 9 describes the fields in the display.

**Table 9 Show IP NHRP Field Descriptions**

Field	Description
100.0.0.2 255.255.255.255	IP address and its network mask in the IP-to-NBMA address cache. The mask is currently always 255.255.255.255 because we do not support aggregation of NBMA information through NHRP.
ATM0/0 created 0:00:43	Interface type and number (in this case, ATM slot and port numbers) and how long ago it was created (hours:minutes:seconds).
expire 1:59:16	Time in which the positive and negative authoritative NBMA address will expire (hours:minutes:seconds). This value is based on the <b>ip nhrp holdtime</b> command.
Type	Value can be one of the following: <ul style="list-style-type: none"> <li>• dynamic—NBMA address was obtained from NHRP Request packet.</li> <li>• static—NBMA address was statically configured.</li> </ul>
Flags	Value can be one of the following: <ul style="list-style-type: none"> <li>• authoritative—Indicates that the NHRP information was obtained from the Next Hop Server or router that maintains the NBMA-to-IP address mapping for a particular destination.</li> <li>• implicit—Indicates that the information was learned not from an NHRP request generated from the local router, but from an NHRP packet being forwarded or from an NHRP request being received by the local router.</li> <li>• negative—For negative caching; indicates that the requested NBMA mapping could not be obtained.</li> </ul>
NBMA address	Nonbroadcast, multiaccess address. The address format is appropriate for the type of network being used (for example, ATM, Ethernet, SMDS, multipoint tunnel).

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

### **ip nhrp map**

## show ip nhrp traffic

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

**show ip nhrp traffic**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Sample Display

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

```
Router# show ip nhrp traffic
Tunnel0
  request packets sent: 2
  request packets received: 4
  reply packets sent: 4
  reply packets received: 2
  register packets sent: 0
  register packets received: 0
  error packets sent: 0
  error packets received: 0
Router#
```

Table 10 describes the fields in the display.

**Table 10 Show IP NHRP Traffic Field Descriptions**

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

## show ip redirects

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

**show ip redirects**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

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

```
Router# 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
Router#
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**ip redirects**

## term ip netmask-format

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

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

### Syntax Description

<b>bitcount</b>	Addresses are followed by a slash and the total number of bits in the netmask. For example, 131.108.11.55/24 indicates that the netmask is 24 bits.
<b>decimal</b>	Netmasks are displayed in dotted decimal notation (for example, 255.255.255.0).
<b>hexadecimal</b>	Netmasks are displayed in hexadecimal format, as indicated by the leading 0X (for example, 0FFFFFF0).

### Default

Netmasks are displayed in dotted decimal format.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

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

However, you can specify that the display of the network mask appear in hexadecimal format or bit count format instead. The hexadecimal format is commonly used on UNIX systems. The previous example would be displayed as 131.108.11.55 0FFFFFF0.

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

### Example

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

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

## trace (privileged)

To discover the routes the packets follow when traveling to their destination from the router, use the **trace** privileged EXEC command.

```
trace [destination]
```

### Syntax Description

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

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The **trace** command works by taking advantage of the error messages generated by the Cisco IOS software 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 router 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 router 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 routers, 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:

```
Router# 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 11 describes the fields shown in the display.

**Table 11 Trace Field Descriptions for IP Routes**

Field	Description
1	Indicates the sequence number of the router in the path to the host.
DEBRIS.CISCO.COM	Host name of this router.
131.108.1.61	Internet address of this router.
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:

```
Router# trace

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

Table 12 describes the fields that are unique to the extended trace sequence, as shown in the display.

**Table 12 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 router to use as a source address for the probes. The Cisco IOS software 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 <b>trace</b> 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 <b>trace</b> command issues prompts for the required fields. Note that <b>trace</b> will place the requested options in each probe; however, there is no guarantee that all routers (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 <b>trace</b> 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 13 describes the characters that can appear in **trace** output.

**Table 13 IP Trace Text Characters**

Character	Description
<i>nn</i> msec	For each node, the round-trip time (in milliseconds) for the specified number of probes.
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable (possibly due to an access list).
Q	Source quench.
P	Protocol unreachable.
N	Network unreachable.
U	Port unreachable.
H	Host unreachable.

**Related Commands**

You can use the master indexes or search online to find documentation of related commands.

**trace (user)**

## trace (user)

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

**trace ip** *destination*

### Syntax Description

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

### Command Mode

User EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The **trace** command works by taking advantage of the error messages generated by the Cisco IOS software 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 router 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 router 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 routers, 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

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

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

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

In the **trace (privileged)** command section, Table 11 describes the fields shown in the display. Table 13 describes the characters that can appear in **trace** output.

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**trace (privileged)**

## tunnel mode

To set the encapsulation mode for the tunnel interface, use the **tunnel mode** interface configuration command. To set to the default, use the **no** form of this command.

```
tunnel mode { aurp | cayman | dvmrp | eon | gre ip [multipoint] | ipip | nos }  
no tunnel mode
```

### Syntax Description

<b>aurp</b>	AppleTalk Update-Based Routing Protocol (AURP).
<b>cayman</b>	Cayman TunnelTalk AppleTalk encapsulation.
<b>dvmrp</b>	Distance Vector Multicast Routing Protocol.
<b>eon</b>	EON compatible CLNS tunnel.
<b>gre ip</b>	Generic routing encapsulation (GRE) protocol over IP.
<b>multipoint</b>	(Optional) Enables a GRE tunnel to be used in a multipoint fashion. Can be used with the <b>gre ip</b> keyword only, and requires the use of the <b>tunnel key</b> command.
<b>ipip</b>	IP over IP encapsulation.
<b>nos</b>	KA9Q/NOS compatible IP over IP.

### Default

GRE tunneling

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0. The **ipip** keyword first appeared in Cisco IOS Release 10.3.

You cannot have two tunnels using the same encapsulation mode with exactly the same source and destination address. The workaround is to create a loopback interface and source packets off of the loopback interface.

Cayman tunneling implements tunneling as designed by Cayman Systems. This enables our routers and access servers to interoperate with Cayman GatorBoxes. With Cayman tunneling, you can establish tunnels between two routers or between our device and a GatorBox. When using Cayman tunneling, you must not configure the tunnel with an AppleTalk network address. This means that there is no way to ping the other end of the tunnel.

Use Distance Vector Multicast Routing Protocol (DVMRP) when a router connects to an mroutered router to run DVMRP over a tunnel. It is required to configure Protocol-Independent Multicast (PIM) and an IP address on a DVMRP tunnel.

Generic routing encapsulation (GRE) tunneling can be done between our routers and access servers only. When using GRE tunneling for AppleTalk, you configure the tunnel with an AppleTalk network address. This means that you can ping the other end of the tunnel.

For multipoint GRE tunnels, a tunnel key must be configured. Unlike other tunnels, the tunnel destination is optional. However, if the tunnel destination is supplied, it must map to an IP multicast address.

### Examples

The following example enables Cayman tunneling:

```
interface tunnel 0
 tunnel source ethernet 0
 tunnel destination 131.108.164.19
 tunnel mode cayman
```

The following example enables GRE tunneling:

```
interface tunnel 0
 appletalk cable-range 4160-4160 4160.19
 appletalk zone Engineering
 tunnel source ethernet 0
 tunnel destination 131.108.164.19
 tunnel mode gre ip
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**appletalk cable-range**  
**appletalk zone**  
**tunnel destination**  
**tunnel source**