# ip accounting through ip sctp authenticate

# ip accounting

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

**ip accounting [access-violations] [output-packets]**

**no ip accounting [access-violations] [output-packets]**

## Syntax Description

| | |
|---|---|
| **access-violations** | (Optional) Enables IP accounting with the ability to identify IP traffic that fails IP access lists. |
| **output-packets** | (Optional) Enables IP accounting based on the IP packets output on the interface. |

## Command Default

IP accounting is disabled on an interface.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 10.3 | The **access-violations** keyword was added. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The **ip accounting** command records the number of bytes (IP header and data) and packets switched through the system on a source and destination IP address basis. Only transit IP traffic is measured and only on an outbound basis; traffic generated by the router access server or terminating in this device is not included in the accounting statistics.

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

To receive a logging message on the console when an extended access list entry denies a packet access (to log violations), you must include the **log** keyword in the **access-list**(IP extended) or **access-list**(IP standard) command.

Statistics are accurate even if IP fast switching or IP access lists are being used on the interface. If the **access-violations** keyword is specified and any IP access list is being used on an interface, then only process switching can generate accurate statistics (IP fast switching or CEF cannot).

IP accounting disables autonomous switching, SSE switching, and distributed switching (dCEF) on the interface. IP accounting will cause packets to be switched on the Route Switch Processor (RSP) instead of the Versatile Interface Processor (VIP), which can cause performance degradation.

## Examples

The following example enables IP accounting on Ethernet interface 0:

```
Router(config)# interface ethernet 0
Router(config-if)# ip accounting
```

## Related Commands

| Command | Description |
|---|---|
| **access-list (IP extended)** | Defines an extended IP access list. |
| **access-list (IP standard)** | Defines a standard IP access list. |
| **clear ip accounting** | Clears the active or checkpointed database when IP accounting is enabled. |
| **ip accounting-list** | Defines filters to control the hosts for which IP accounting information is kept. |
| **ip accounting-threshold** | Sets the maximum number of accounting entries to be created. |
| **ip accounting-transits** | Controls the number of transit records that are stored in the IP accounting database. |
| **show ip accounting** | Displays the active accounting or checkpointed database or displays access list violations. |

# ip accounting-list

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

**ip accounting-list** *ip-address wildcard*

**no ip accounting-list** *ip-address wildcard*

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address in dotted decimal format. |
| *wildcard* | Wildcard bits to be applied to the *ip-address*argument. |

## Command Default

No filters are defined.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The *wildcard* argument is a 32-bit quantity written in dotted-decimal format. Address bits corresponding to wildcard bits set to 1 are ignored in comparisons; address bits corresponding to wildcard bits set to zero are used in comparisons.

## Examples

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

```
Router(config)# ip accounting-list 192.31.0.0 0.0.255.255
```

## Related Commands

| Command | Description |
|---|---|
| **clear ip accounting** | Clears the active or checkpointed database when IP accounting is enabled. |

| Command | Description |
|---|---|
| **ip accounting** | Enables IP accounting on an interface. |
| **ip accounting-threshold** | Sets the maximum number of accounting entries to be created. |
| **ip accounting-transits** | Controls the number of transit records that are stored in the IP accounting database. |
| **show ip accounting** | Displays the active accounting or checkpointed database or displays access list violations. |

# ip accounting mac-address

To enable IP accounting on a LAN interface based on the source and destination Media Access Control (MAC) address, use the **ip accounting mac-address** command in interface configuration mode. To disable IP accounting based on the source and destination MAC address, use the **no** form of this command.

**ip accounting mac-address** {**input**| **output**}

**no ip accounting mac-address** {**input**| **output**}

## Syntax Description

| | |
|---|---|
| **input** | Performs accounting based on the source MAC address on received packets. |
| **output** | Performs accounting based on the destination MAC address on transmitted packets. |

## Command Default

IP accounting is disabled on an interface.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 11.1CC | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |
| 12.2(33)SCB | This command was integrated into Cisco IOS Release 12.2(33)SCB. |

## Usage Guidelines

This feature is supported on Ethernet, Fast Ethernet, and FDDI interfaces.

To display the MAC accounting information, use the **show interface mac** EXEC command.

MAC address accounting provides accounting information for IP traffic based on the source and destination MAC address on LAN interfaces. This calculates the total packet and byte counts for a LAN interface that receives or sends IP packets to or from a unique MAC address. It also records a timestamp for the last packet received or sent. With MAC address accounting, you can determine how much traffic is being sent to and/or received from various peers at NAPS/peering points.

## Examples

The following example enables IP accounting based on the source and destination MAC address for received and transmitted packets:

```
Router(config)# interface ethernet 4/0/0
Router(config-if)# ip accounting mac-address input
Router(config-if)# ip accounting mac-address output
```

## Examples

The following example enables IP accounting based on the source MAC address for received packets on a Gigabit Ethernet interface:

```
Router# configure terminal
Router(config)# interface GigabitEthernet3/0/0
Router(config-if)# ip accounting mac-address input
```

## Related Commands

| Command | Description |
|---|---|
| **show interface mac** | Displays MAC accounting information for interfaces configured for MAC accounting. |

# ip accounting precedence

To enable IP accounting on any interface based on IP precedence, use the **ip accounting precedence** command in interface configuration mode. To disable IP accounting based on IP precedence, use the **no** form of this command.

**ip accounting precedence** {**input**| **output**}

**no ip accounting precedence** {**input**| **output**}

## Syntax Description

| | |
|---|---|
| **input** | Performs accounting based on IP precedence on received packets. |
| **output** | Performs accounting based on IP precedence on transmitted packets. |

## Command Default

IP accounting is not enabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 11.1CC | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

To display IP precedence accounting information, use the **show interface precedence** EXEC command.

The precedence accounting feature provides accounting information for IP traffic, summarized by IP precedence values. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on IP precedence. This feature is supported on all interfaces and subinterfaces and supports Cisco Express Forwarding (CEF), dCEF, flow, and optimum switching.

## Examples

The following example enables IP accounting based on IP precedence for received and transmitted packets:

```
Router(config)# interface ethernet 4/0/0
Router(config-if)# ip accounting precedence input
Router(config-if)# ip accounting precedence output
```

## Related Commands

| Command | Description |
|---|---|
| **show interface precedence** | Displays precedence accounting information for an interface configured for precedence accounting. |

# ip accounting-threshold

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

**ip accounting-threshold** *threshold*

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

## Syntax Description

| | |
|---|---|
| *threshold* | Maximum number of entries (source and destination address pairs) that the Cisco IOS software accumulates. |

## Command Default

The default maximum number of accounting entries is 512 entries.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

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

The default accounting threshold of 512 entries results in a maximum table size of 12,928 bytes. Active and checkpointed tables can reach this size independently.

## Examples

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

```
Router(config)# ip accounting-threshold 500
```

## Related Commands

| Command | Description |
|---|---|
| **clear ip accounting** | Clears the active or checkpointed database when IP accounting is enabled. |
| **ip accounting** | Enables IP accounting on an interface. |
| **ip accounting-list** | Defines filters to control the hosts for which IP accounting information is kept. |
| **ip accounting-transits** | Controls the number of transit records that are stored in the IP accounting database. |
| **show ip accounting** | Displays the active accounting or checkpointed database or displays access list violations. |

# ip accounting-transits

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

**ip accounting-transits** *count*

**no ip accounting-transits**

## Syntax Description

| | |
|---|---|
| *count* | Number of transit records to store in the IP accounting database. |

## Command Default

The default number of transit records that are stored in the IP accounting database is 0.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

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

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

## Examples

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

```
Router(config)# ip accounting-transits 100
```

## Related Commands

| Command | Description |
|---|---|
| **clear ip accounting** | Clears the active or checkpointed database when IP accounting is enabled. |
| **ip accounting** | Enables IP accounting on an interface. |
| **ip accounting-list** | Defines filters to control the hosts for which IP accounting information is kept. |
| **ip accounting-threshold** | Sets the maximum number of accounting entries to be created. |
| **show ip accounting** | Displays the active accounting or checkpointed database or displays access list violations. |

# 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. |

## Command Default

Default address: 255.255.255.255 (all ones)

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Examples

The following example specifies an IP broadcast address of 0.0.0.0:

```
Router(config-if)# ip broadcast-address 0.0.0.0
```

# ip casa

To configure the router to function as a forwarding agent, use the **ip casa** command in global configuration mode. To disable the forwarding agent, use the **no** form of this command.

**ip casa** *control-address igmp-address* [ *udp-limit* ]

**no ip casa**

## Syntax Description

| | |
|---|---|
| *control-address* | IP address of the forwarding agent side of the services manager and forwarding agent tunnel used for sending signals. This address is unique for each forwarding agent. |
| *igmp-address* | Interior Gateway Management Protocol (IGMP) address on which the forwarding agent will listen for wildcard and fixed affinities. |
| *udp-limit* | (Optional) Maximum User Datagram Protocol (UDP) queue length; valid values are from 50 to 65535. The default is 256. |

## Command Default

No default behavior or values.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)T | This command was introduced. |
| 12.2(17d)SXB1 | Support for this command was added for Catalyst 6500 series switches. |
| 12.2(18)SXF6 | The *udp-limit* argument was added. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |

## Usage Guidelines

If more than the maximum *udp-limit* value arrives in a burst, the Cisco Appliance Services Architecture (CASA) wildcard updates from the service manager might get dropped.

The *control-address* value is unique for each forwarding agent.

## Examples

The following example specifies the Internet address (10.10.4.1) and IGMP address (224.0.1.2) for the forwarding agent and sets the UDP queue length to 300:

```
Router(config)# ip casa 10.10.4.1 224.0.1.2 300
```

## Related Commands

| Command | Description |
|---|---|
| **forwarding-agent** | Specifies the port on which the forwarding agent will listen for wildcard and fixed affinities. |

# ip cef traffic-statistics

To change the time interval that controls when Next Hop Resolution Protocol (NHRP) sets up or tears down a switched virtual circuit (SVC), use the **ip cef traffic-statistics** command in global configuration mode. To restore the default values, use the **no** form of this command.

**ip cef traffic-statistics** [**load-interval** *seconds*] [**update-rate** *seconds*]

**no ip cef traffic-statistics**

## Syntax Description

| | |
|---|---|
| **load-interval** *seconds* | (Optional) Length of time (in 30-second increments) during which the average *trigger-threshold* and *teardown-threshold* intervals are calculated before an SVC setup or teardown action is taken. (These thresholds are configured in the **ip nhrp trigger-svc** command.) The **load-interval** range is from 30 seconds to 300 seconds, in 30-second increments. The default value is 30 seconds. |
| **update-rate** *seconds* | (Optional) Frequency that the port adapter sends the accounting statistics to the Route Processor (RP). When the route processor is using NHRP in distributed Cisco Express Forwarding switching mode, this value must be set to 5 seconds. The default value is 10 seconds. |

## Command Default

Load interval: 30 seconds Update rate: 10 seconds

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The **ip nhrp trigger-svc** command sets the threshold by which NHRP sets up and tears down a connection. The threshold is the Cisco Express Forwarding traffic load statistics. The thresholds in the **ip nhrp trigger-svc**

command are measured during a sampling interval of 30 seconds, by default. To change that interval over which that threshold is determined, use the **load-interval** *seconds* option of the **ip cef traffic-statistics** command.

When NHRP is configured on a Cisco Express Forwarding switching node with a Versatile Interface Processor (VIP2) adapter, you must make sure the **update-rate** keyword is set to 5 seconds.

Other Cisco IOS features could also use the **ip cef traffic-statistics** command; this NHRP feature relies on it.

## Examples

In the following example, the triggering and teardown thresholds are calculated based on an average over 120 seconds:

```
Router(config)# ip cef traffic-statistics load-interval 120
```

## Related Commands

| Command | Description |
|---|---|
| **ip nhrp trigger-svc** | Configures when NHRP will set up and tear down an SVC based on aggregate traffic rates. |

# ip directed-broadcast

To enable the translation of a 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*| *extended access-list-number*]

**no ip directed-broadcast** [*access-list-number*| *extended access-list-number*]

## Syntax Description

| | |
|---|---|
| *access-list-number* | (Optional) Standard access list number in the range from 1 to 199. If specified, a broadcast must pass the access list to be forwarded. |
| *extended access-list-number* | (Optional) Extended access list number in the range from 1300 to 2699. |

## Command Default

Disabled; all IP directed broadcasts are dropped.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.0 | The default behavior changed to directed broadcasts being dropped. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |
| Cisco IOS XE 3.3SE | This command was implemented in Cisco IOS XE Release 3.3SE. |

## Usage Guidelines

An IP directed broadcast is an IP packet whose destination address is a valid broadcast address for some IP subnet, but which originates from a node that is not itself part of that destination subnet.

A router that is not directly connected to its destination subnet forwards an IP directed broadcast in the same way it would forward unicast IP packets destined to a host on that subnet. When a directed broadcast packet reaches a router that is directly connected to its destination subnet, that packet is “exploded” as a broadcast on the destination subnet. The destination address in the IP header of the packet is rewritten to the configured IP broadcast address for the subnet, and the packet is sent as a link-layer broadcast.

The **ip directed-broadcast** command controls the explosion of directed broadcasts when they reach their target subnets. The command affects only the final transmission of the directed broadcast on its ultimate destination subnet. It does not affect the transit unicast routing of IP directed broadcasts.

If **directed broadcast** is enabled for an interface, incoming IP packets whose addresses identify them as directed broadcasts intended for the subnet to which that interface is attached will be exploded as broadcasts on that subnet. If an access list has been configured with the **ip directed-broadcast** command, only directed broadcasts that are permitted by the access list in question will be forwarded; all other directed broadcasts destined for the interface subnet will be dropped.

If the **no ip directed-broadcast** command has been configured for an interface, directed broadcasts destined for the subnet to which that interface is attached will be dropped, rather than being broadcast.

**Note** Because directed broadcasts, and particularly Internet Control Message Protocol (ICMP) directed broadcasts, have been abused by malicious persons, we recommend that security-conscious users disable the **ip directed-broadcast** command on any interface where directed broadcasts are not needed and that they use access lists to limit the number of exploded packets.

## Examples

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

```
Router(config)# interface ethernet 0
Router(config-if)# ip directed-broadcast
```

## Related Commands

| Command | Description |
|---|---|
| **ip forward-protocol** | Specifies which protocols and ports the router forwards when forwarding broadcast packets. |

# ip forward-protocol

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

**ip forward-protocol** {**udp** [ *port* ]| **nd**| **sdns**}

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

## Syntax Description

| | |
|---|---|
| **udp** | Forwards User Datagram Protocol (UDP) packets. See the "Usage Guidelines" section for a list of port numbers forwarded by default. |
| *port* | (Optional) Destination port that controls which UDP services are forwarded. |
| **nd** | Forwards Network Disk (ND) packets. This protocol is used by older diskless Sun workstations. |
| **sdns** | Secure Data Network Service. |

## Command Default

Router forwarding is enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

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, Routing Information Protocol (RIP)] may be hazardous to your network.

If you use the **ip forward-protocol** command, specifying only 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. Because 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.

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 packets (ports 67 and 68)
- TACACS service (port 49)
- IEN-116 Name Service (port 42)

**Note** If UDP port 68 is used as the destination port number, it is not forwarded by default.

## Examples

The following example defines a helper address and uses the **ip forward-protocol** command. Using the **udp** keyword without specifying any port numbers will allow forwarding of UDP packets on the default ports.

```
Router(config)# ip forward-protocol udp
Router(config)# interface ethernet 1
Router(config-if)# ip helper-address 10.24.42.2
```

# 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** command in global configuration mode. To disable the flooding of IP broadcasts, use the **no** form of this command.

**ip forward-protocol spanning-tree [any-local-broadcast]**

**no ip forward-protocol spanning-tree [any-local-broadcast]**

## Syntax Description

| | |
|---|---|
| **any-local-broadcast** | (Optional) Accept any local broadcast when flooding. |

## Command Default

IP broadcast flooding is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

A packet must meet the following criteria to be considered for flooding:

- The MAC address of the received frame must be all-ones broadcast address (ffff.ffff.ffff).
- The IP destination address must be one of the following: all-ones broadcast (255.255.255.255), subnet broadcast for the receiving interface; major-net broadcast for the receiving interface if the **no ip classless** command is also configured; or any local IP broadcast address if the **ip forward-protocol spanning-tree any-local-broadcast** command is configured.
- The IP time-to-live (TTL) value must be at least 2.
- The IP protocol must be User Datagram Protocol (UDP) (17).
- The UDP destination port must be TFTP, Domain Name System (DNS), Time, NetBIOS, ND, or BOOTP packet, or a UDP port specified by the **ip forward-protocol udp** command.

A flooded UDP datagram is given the destination address specified by the **ip broadcast-address** 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 *Cisco IOS 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 (10.108.255.255 as an example in the network number 10.108.0.0). This mechanism also forward packets whose contents are the zeros version of the all-networks broadcast when subnetting is enabled (for example, 10.108.0.0).

This command is an extension of the **ip helper-address** 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.

## Examples

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

```
Router(config)# ip forward-protocol spanning-tree
```

## Related Commands

| Command | Description |
|---|---|
| **ip broadcast-address** | Defines a broadcast address for an interface. |
| **ip forward-protocol** | Specifies which protocols and ports the router forwards when forwarding broadcast packets. |
| **ip forward-protocol turbo-flood** | Speeds up flooding of UDP datagrams using the spanning-tree algorithm. |
| **ip helper-address** | Forwards UDP broadcasts, including BOOTP, received on an interface. |

# 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** command in global configuration mode. To disable this feature, use the **no**form of this command.

**ip forward-protocol turbo-flood [udp-checksum]**

**no ip forward-protocol turbo-flood [udp-checksum]**

## Syntax Description

| | |
|---|---|
| **udp-checksum** | (Optional) UDP checksum. |

## Command Default

UDP turbo flooding is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(17d)SXB7 | Support for this command was introduced on the Supervisor Engine 720. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |

## Usage Guidelines

Used in conjunction with the **ip forward-protocol spanning-tree** command, this command is supported over Advanced Research Projects Agency (ARPA)-encapsulated Ethernets, FDDI, and High-Level Data Link Control (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.

When you enter the **ip forward-protocol turbo-flood** command, the outgoing UDP packets have a NULL checksum. If you want to have UDP checksums on all outgoing packets, you must enter the **ip forward-protocol turbo-flood udp-checksum** command.

## Examples

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

```
Router(config)# ip forward-protocol turbo-flood
Router(config)# ip forward-protocol spanning-tree
!
Router(config)# interface ethernet 0
Router(config-if)# ip address 10.9.1.1
Router(config-if)# bridge-group 1
```

```
!
Router(config)# interface ethernet 1
Router(config-if)# ip address 10.9.1.2
Router(config-if)# bridge-group 1
!
Router(config)# bridge 1 protocol dec
```

The following example shows how to speed up the flooding of UDP packets using the spanning-tree algorithm and include the UDP checksums on all outgoing packets:

```
Router(config)# ip forward-protocol turbo-flood udp-checksum
```

## Related Commands

| Command | Description |
|---|---|
| **ip forward-protocol** | Specifies which protocols and ports are forwarded by the router when forwarding broadcast packets. |
| **ip forward-protocol spanning-tree** | Permits IP broadcasts to be flooded throughout the internetwork in a controlled fashion. |

# ip header-compression special-vj

To enable the special Van Jacobson (VJ) format of TCP header compression, use the **ip header-compression special-vj** command in interface configuration mode. To disable the special VJ format and return to the default VJ format, use the **no** form of this command.

**ip header-compression special-vj**

**no ip header-compression special-vj**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The default VJ format of TCP header compression is enabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.4(15)T12 | This command was introduced. |
| 15.0(1)M2 | This command was integrated into Cisco IOS Release 15.0(1)M2. |

## Usage Guidelines

Use the **ip tcp header-compression** command to enable the default VJ format of TCP header compression. Then use the **ip header-compression special-vj** command to enable the special VJ format of TCP header compression.

To enable the special VJ format of TCP header compression so that context IDs are included in compressed packets, use the **special-vj** command in IPHC profile configuration mode.

## Examples

The following example shows how to configure the special VJ format of TCP header compression for serial interface 5/0:

```
Router(config)# interface serial 5/0
Router(config-if)# ip header-compression special-vj

Building configuration...
Current configuration : 579 bytes
!
interface Serial 5/0
 bandwidth 4032
 ip address 10.72.72.3 255.255.255.0
 encapsulation frame-relay
 shutdown
 no keepalive
 serial restart-delay 0
 no arp frame-relay
 frame-relay map ip 10.72.72.2 100 broadcast
 frame-relay ip tcp header-compression
```

```
 frame-relay ip tcp compression-connections 8
 frame-relay ip rtp header-compression periodic-refresh
 frame-relay ip rtp compression-connections 8
 service-policy output p1
 ip header-compression special-vj
 ip header-compression max-header 60
 ip header-compression max-time 50
 ip header-compression max-period 32786
end
```

## Related Commands

| Command | Description |
|---|---|
| **ip tcp header-compression** | Enables TCP header compression. |
| **show ip tcp header-compression** | Displays TCP/IP header compression statistics. |
| **special-vj** | Enables the special VJ format of TCP header compression so that context IDs are included in compressed packets. |

# ip helper-address

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

**ip helper-address**[**vrf** *name*| **global**] *address* {[**redundancy** *vrg-name*]}

**no ip helper-address** [**vrf** *name*| **global**] *address* {[**redundancy** *vrg-name*]}

## Syntax Description

| | |
|---|---|
| **vrf** *name* | (Optional) Enables the VPN routing and forwarding (VRF) instance and the VRF name. |
| **global** | (Optional) Configures a global routing table. |
| *address* | Destination broadcast or host address to be used when forwarding UDP broadcasts. There can be more than one helper address per interface. |
| **redundancy** *vrg-name* | (Optional) Defines the Virtual Router Group (VRG) name. |

## Command Default

UDP broadcasts are not forwarded.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(4)B | This command was modified. The **vrf** *name* keyword and argument pair and the **global** keyword were added. |
| 12.2(15)T | This command was modified. The **redundancy** *vrg-name* keyword and argument pair was added. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The **ip forward-protocol** command along with the **ip helper-address** command allows you to control broadcast packets and protocols that are forwarded.

One common application that requires helper addresses is DHCP, which is defined in RFC 1531. To enable BOOTP or DHCP broadcast forwarding for a set of clients, configure a helper address on the router interface connected to the client. The helper address must specify the address of the BOOTP or DHCP server. If you have multiple servers, configure one helper address for each server.

The following conditions must be met for a UDP or IP packet to be able to use the **ip helper-address** command:

- The MAC address of the received frame must be all-ones broadcast address (ffff.ffff.ffff).
- The IP destination address must be one of the following: all-ones broadcast (255.255.255.255), subnet broadcast for the receiving interface, or major-net broadcast for the receiving interface if the **no ip classless** command is also configured.
- The IP time-to-live (TTL) value must be at least 2.
- The IP protocol must be UDP (17).
- The UDP destination port must be for TFTP, Domain Name System (DNS), Time, NetBIOS, ND, BOOTP or DHCP packet, or a UDP port specified by the **ip forward-protocol udp** command in global configuration mode.

If the DHCP server resides in a VPN or global space that is different from the interface VPN, then the **vrf** *name* or the **global** option allows you to specify the name of the VRF or global space in which the DHCP server resides.

The **ip helper-addressvrf***name address* option uses the address associated with the VRF name regardless of the VRF of the incoming interface. If the **ip helper-addressvrf***name address* command is configured and later the VRF is deleted from the configuration, then all IP helper addresses associated with that VRF name will be removed from the interface configuration.

If the **ip helper-address** *address* command is already configured on an interface with no VRF name configured, and later the interface is configured with the **ip helper-address vrf** *name address* command, then the previously configured **ip helper-address** *address* command is considered to be global.

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

The **service dhcp** command must be configured on the router to enable IP helper statements to work with DHCP. If the command is not configured, the DHCP packets will not be relayed through the IP helper statements. The **service dhcp** command is configured by default.

## Examples

The following example shows how to define an address that acts as a helper address:

```
Router(config)# interface ethernet 1
Router(config-if)# ip helper-address 10.24.43.2
```

The following example shows how to define an address that acts as a helper address and is associated with a VRF named host1:

```
Router(config)# interface ethernet 1/0
Router(config-if)# ip helper-address vrf host1 10.25.44.2
```

The following example shows how to define an address that acts as a helper address and is associated with a VRG named group1:

```
Router(config)# interface ethernet 1/0
Router(config-if)# ip helper-address 10.25.45.2 redundancy group1
```

## Related Commands

| Command | Description |
|---|---|
| **ip forward-protocol** | Specifies which protocols and ports the router forwards when forwarding broadcast packets. |
| **service dhcp** | Enables the DHCP server and relay agent features on the router. |

# ip icmp rate-limit unreachable

To limit the rate at which Internet Control Message Protocol (ICMP) unreachable messages are generated for a destination, use the **ip icmp rate-limit unreachable** command in global configuration mode. To use the default, use the **no** form of this command.

**ip icmp rate-limit unreachable [df]** [ *ms* ] [**log** [ *packets* ] [ *interval-ms* ]]

**no ip icmp rate-limit unreachable [df]** [ *ms* ] [**log** [ *packets* ] [ *interval-ms* ]]

## Syntax Description

| | |
|---|---|
| **df** | (Optional) Don't Fragment (DF) bit is set. The optional *ms* argument is a time limit in milliseconds (ms) in which one unreachable message is generated. If the **df** keyword is specified, its *ms* argument remains independent from those of general destination unreachable messages.<br>The valid range is from 1 ms to 4294967295 ms.<br>**Note** Counting begins as soon as this command is configured. |
| **log** | (Optional) Logging of generated messages that show packets that could not reach a destination at a specified threshold. The optional *packets* argument specifies a packet threshold. When it is reached, a log message is generated on the console. The default is 1000 packets. The optional *interval-ms* argument is a time limit for the interval for which a logging message is triggered. The default is 60000 ms, which is 1 minute. |

## Command Default

The default value is one ICMP destination unreachable message per 500 ms.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.0 | This command was introduced. |
| 12.4(2)T | The *packets* and the *interval-ms* arguments and **log** keyword were introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(31)SB2 | This command was integrated into Cisco IOS Release 12.2(31)SB2. |

| Release | Modification |
|---|---|
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

Counting of packets begins when the command is configured and a packet threshold is specified.

The **no ip icmp rate-limit unreachable** command turns off the previously configured rate limit. To reset the rate limit to its default value, use the **ip icmp rate-limit unreachable** command default.

Cisco IOS software maintains two timers: one for general destination unreachable messages and one for DF destination unreachable messages. Both share the same time limits and defaults. If the **df** option is not configured, the **ip icmp rate-limit unreachable** command sets the time values in ms for DF destination unreachable messages.

## Examples

The following example sets the rate of the ICMP destination unreachable message to one message every 10 ms:

```
Router(config)# ip icmp rate-limit unreachable 10
```

The following example turns off the previously configured rate limit:

```
Router(config)# no ip icmp rate-limit unreachable
```

The following example sets the rate limit back to the default:

```
Router(config)# no ip icmp rate-limit unreachable
```

The following example sets a logging packet threshold and time interval:

```
Router(config)# ip icmp rate-limit unreachable log 1200 120000
```

## Related Commands

| Command | Description |
|---|---|
| **clear ip icmp rate-limit** | Clears all ICMP unreachable destination messages or all statistics for a specified interface. |
| **show ip icmp rate-limit** | Displays all ICMP unreachable destination messages or all statistics for a specified interface. |

# ip icmp redirect

To control the type of Internet Control Message Protocol (ICMP) redirect message that is sent by the Cisco IOS software, use the **ip icmp redirect** command in global configuration mode. To set the value back to the default, use the **no**form of this command.

**ip icmp redirect** [**host**| **subnet**]

**no ip icmp redirect** [**host**| **subnet**]

## Syntax Description

| | |
|---|---|
| **host** | (Optional) Sends ICMP host redirects. |
| **subnet** | (Optional) Sends ICMP subnet redirects. |

## Command Default

The router will send ICMP subnet redirect messages.

Because the **ip icmp redirect subnet** command is the default, the command will not be displayed in the configuration.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

An ICMP redirect message can be generated by a router when a packet is received and transmitted on the same interface. In this situation, the router will forward the original packet and send a ICMP redirect message back to the sender of the original packet. This behavior allows the sender to bypass the router and forward future packets directly to the destination (or a router closer to the destination).

There are two types of ICMP redirect messages: redirect for a host address or redirect for an entire subnet.

The **ip icmp redirect** command determines the type of ICMP redirects sent by the system and is configured on a per system basis. Some hosts do not understand ICMP subnet redirects and need the router to send out ICMP host redirects. Use the **ip icmp redirect host** command to have the router send out ICMP host redirects. Use the **ip icmp redirect subnet** command to set the value back to the default, which is to send subnet redirects.

To prevent the router from sending ICMP redirects, use the **no ip redirects**interface configuration command.

## Examples

The following example enables the router to send out ICMP host redirects:

```
Router(config)# ip icmp redirect host
```

The following example sets the value back to the default, which is subnet redirects:

```
Router(config)# ip icmp redirect subnet
```

## Related Commands

| Command | Description |
|---|---|
| **ip redirects** | Enables the sending of ICMP redirect messages. |

# ip information-reply

To configure Cisco IOS software to send Internet Control Message Protocol (ICMP) information replies, use the **ip information-reply** command in interface configuration mode. To disable this function, use the **no** form of this command.

**ip information-reply**

**no ip information-reply**

## Syntax Description

This command has no arguments or keywords.

## Command Default

ICMP information replies are not sent.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2T | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

The ability for the Cisco IOS software to respond to ICMP information request messages with an ICMP information reply message is disabled by default. Use this command to allow the software to send ICMP information reply messages.

## Examples

The following example enables the sending of ICMP information reply messages on Ethernet interface 0:

```
Router(config)# interface ethernet 0
Router(config-if)# ip address 10.108.1.0 255.255.255.0
Router(config-if)# ip information-reply
```

# ip mask-reply

To configure Cisco IOS software to respond to Internet Control Message Protocol (ICMP) mask requests by sending ICMP mask reply messages, use the **ip mask-reply** command in interface configuration mode. To disable this function, use the no form of this command.

**ip mask-reply**

**no ip mask-reply**

## Syntax Description

This command has no arguments or keywords.

## Command Default

ICMP mask reply messages are not sent.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Examples

The following example enables the sending of ICMP mask reply messages on Ethernet interface 0:

```
Router(config)# interface ethernet 0
Router(config-if)# ip address 10.108.1.0 255.255.255.0
Router(config-if)# ip mask-reply
```

# ip mtu

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

**ip mtu** *bytes*

**no ip mtu**

## Syntax Description

| | |
|---|---|
| *bytes* | MTU size, in bytes. |

## Command Default

The default MTU value depends on the interface type.

***Table 1: Default MTU Values by Interface Type***

| Interface Type | Default MTU (Bytes) |
|---|---|
| ATM | 4470 |
| Ethernet | 1500 |
| FDDI | 4470 |
| High-Speed Serial Interface High Speed Access (HSSI HSA) | 4470 |
| Serial | 1500 |
| Token Ring | 4464 |
| VRF-Aware Service Infrastructure (VASI) | 9216 |

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

| Release | Modification |
|---|---|
| Cisco IOS XE Release 2.4 | This command was integrated into Cisco IOS XE Release 2.4. |

## Usage Guidelines

If an IP packet exceeds the MTU size that is set for the interface, the Cisco software fragments the IP packet.

When an IPsec MTU is less than 256 bytes, the crypto engine MTU is set to 256 bytes and packets greater than 256 bytes are fragmented.

For VASI interfaces that involve Ethernet type interfaces (Ethernet, Fast Ethernet, or Gigabit Ethernet), the IP MTU size of a VASI interface must be set to the same value as the lower default setting of the Ethernet type interface of 1500 bytes. If this adjustment is not made, OSPF reconvergence on the VASI interface requires a long time.

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

If a dynamic virtual tunnel interface (VTI) configured with an IP MTU causes encapsulating security payload (ESP) fragmentation, clear and re-establish the encryption session.

When a loopback interface is used as the VTI tunnel source, you must manually configure the **ip mtu** command. This is because the IPsec encapsulation bytes are calculated based on the outgoing physical interface.

**MTU Size in an IPsec Configuration**

In an IPsec configuration, such as in a crypto environment, an MTU value that is less than 256 bytes is not accepted. If you configure an MTU value less than 256 bytes, the MTU value is automatically overwritten and given a value of 256 bytes.

**MTU Size in Cisco ME 3600X Series Ethernet Access Switches**

In Cisco ME 3600X Series Ethernet Access Switches, you can configure seven unique MTU sizes on router and switchport interfaces and eight unique sizes on VLAN interfaces. This does not include the default size of 1500.

## Examples

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

```
Device(config)# interface serial 0
Device(config-if)# ip mtu 300
```

## Related Commands

| Command | Description |
|---|---|
| **mtu** | Adjusts the MTU size. |

# ip redirects

To enable the sending of Internet Control Message Protocol (ICMP) 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** command in interface configuration mode. 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.

## Command Default

ICMP redirect messages are sent.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

Previously, if the Hot Standby Router Protocol (HSRP) was configured on an interface, ICMP redirect messages were disabled by default for the interface. With Cisco IOS Release 12.1(3)T, ICMP redirect messages are enabled by default if HSRP is configured.

## Examples

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

```
Router(config)# interface ethernet 0
Router(config-if)# ip redirects
```

## Related Commands

| Command | Description |
|---|---|
| **ip default-gateway** | Defines a default gateway (router) when IP routing is disabled. |

| Command | Description |
| --- | --- |
| **show ip redirects** | Displays the address of a default gateway (router) and the address of hosts for which an ICMP redirect message has been received. |

# ip sctp asconf

To enable the ability of an existing Stream Control Transmission Protocol (SCTP) endpoint to automatically send Address Configuration Change (ASCONF) chunks in response to an IP address change on a router without an authentication check, use the **ip sctp asconf** command in global configuration mode. To disable the requirement for ASCONF and ASCONF Acknowledgement (ASCONF-ACK) chunks to perform an authentication requirement check, use the **no** form of this command.

**ip sctp asconf** {**authenticate check**| **auto**}

**no ip sctp asconf** {**authenticate check**| **auto**}

## Syntax Description

| | |
|---|---|
| **authenticate check** | Configures SCTP to check that authentication is supported on the endpoint before sending an ASCONF chunk. |
| **auto** | Configures SCTP to automatically send ASCONF chunks in response to an IP address change on a router. |

## Command Default

SCTP checks the authentication status of the endpoint before sending an ASCONF chunk in response to an IP address change on the router.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.4(20)T | This command was introduced. |

## Usage Guidelines

The ASCONF chunk format requires the receiving SCTP to not report to the sender if it does not understand the ASCONF chunk. This command enables you to configure sending the ASCONF chunk automatically in response to an IP address change in an SCTP stream, or to authenticate the endpoint before sending the ASCONF chunk.

The ASCONF chunk is used to communicate to the endpoint of an SCTP stream that at least one of the configuration change requests in the stream must be acknowledged.

## Examples

The following example shows how to configure SCTP to authenticate the endpoint before sending an ASCONF chunk:

```
Router(config)# ip sctp asconf authenticate check
```

The following example shows how to configure SCTP to automatically send an ASCONF chunk in response to a change in the IP address of the remote endpoint:

```
Router(config)# ip sctp asconf auto
```

## Related Commands

| Command | Description |
|---|---|
| **ip sctp authenticate** | To define Stream Control Transmission Protocol (SCTP) data chunks that the client requires be authenticated. |

# ip sctp authenticate

To define Stream Control Transmission Protocol (SCTP) data chunks that the client requires be authenticated, use the **ip sctp authenticate** command in global configuration mode. To disable the authentication of an SCTP data chunk, use the **no** form of this command.

**ip sctp authenticate** {*chunk-type*| *chunk-number*}

**no ip sctp authenticate** {*chunk-type*| *chunk-number*}

## Syntax Description

| | |
|---|---|
| *chunk-type* | Name of the chunk type to be authenticated. See Table 1 in the "Usage Guidelines" section for a list of chunk types. |
| chunk-number | Number of the chunk to be authenticated in the range from 0 to 255. |

## Command Default

SCTP data chunks are not authenticated by default.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.4(15)T | This command was introduced. |
| 12.4(20)T | This command was enhanced to support the Address Configuration (ASCONF) and ASCONF-ACK SCTP chunk types. |

## Usage Guidelines

SCTP Authentication procedures use either Message Digest 5 (MD5) or Secure Hash Algorithm 1 (SHA-1), which can be memory and CPU intensive. Enabling SCTP Authentication on data chunks could impact CPU utilization when a large number of authenticated chunks are sent.

You cannot disable the authentication of the ASCONF or ASCONF-ACK chunks.

Enabling the authentication of a chunk type applies only to new endpoints and associations.

The table below provides a list of SCTP chunk types and SCTP chunk numbers.

***Table 2: SCTP Authentication Chunk Types***

| SCTP Chunk Type | SCTP Chunk Number | Description |
|---|---|---|
| abort association | 0x06 | ABORT chunk. |

| SCTP Chunk Type | SCTP Chunk Number | Description |
|---|---|---|
| asconf | 0xC1 | ASCONF chunk. |
| asconf-ack | 0x80 | ASCONF acknowledgement chunk. |
| cookie-ack | 0x0b | COOKIE acknowledgment chunk. |
| cookie-echo | 0x0a | COOKIE-ECHO chunk. |
| data | 0x00 | DATA chunk. |
| fwd-tsn | 0xc0 | FWD-CUM-TSN chunk. Forwarded cumulative transmission sequence number chunk. |
| heartbeat | 0x04 | HEARTBEAT request chunk. |
| heartbeat-ack | 0x05 | HEARTBEAT acknowledgement chunk. |
| packet-drop | 0x81 | PACKET-DROP chunk. |
| sack | 0x03 | Selective acknowledgment chunk. |
| shutdown | 0x07 | SHUTDOWN chunk. |
| shutdown-ack | 0x08 | SHUTDOWN acknowledgment chunk. |
| stream-reset | 0x82 | STREAM-RESET chunk. |

## Examples

The following example shows how to enable authentication of SCTP data chunks:

```
Router(config)# ip sctp authenticate data
```

## Related Commands

| Command | Description |
|---|---|
| **show sctp association** | Displays accumulated information for a specific SCTP association. |
| **show sctp errors** | Displays the error counts logged by SCTP. |
| **show sctp statistics** | Displays the overall statistics counts for SCTP activity. |