ip inspect through ip security strip

- ip inspect, on page 3
- ip inspect alert-off, on page 5
- ip inspect audit-trail, on page 6
- ip inspect dns-timeout, on page 8
- ip inspect hashtable, on page 10
- ip inspect L2-transparent dhcp-passthrough, on page 11
- ip inspect log drop-pkt, on page 13
- ip inspect max-incomplete high, on page 16
- ip inspect max-incomplete low, on page 18
- ip inspect name, on page 20
- ip inspect one-minute high, on page 33
- ip inspect one-minute low, on page 35
- ip inspect tcp block-non-session, on page 37
- ip inspect tcp finwait-time, on page 39
- ip inspect tcp idle-time, on page 41
- ip inspect tcp max-incomplete host, on page 43
- ip inspect tcp reassembly, on page 45
- ip inspect tcp synwait-time, on page 47
- ip inspect tcp window-scale-enforcement loose, on page 48
- ip inspect udp idle-time, on page 50
- ip inspect waas enable, on page 52
- integrity, on page 53
- ip interface, on page 55
- ip ips, on page 57
- ip ips auto-update, on page 59
- ip ips config location, on page 61
- ip ips deny-action ips-interface, on page 63
- ip ips enable-clidelta, on page 65
- ip ips event-action-rules, on page 66
- ip ips fail closed, on page 67
- ip ips inherit-obsolete-tunings, on page 68
- ip ips memory regex chaining, on page 70
- ip ips memory threshold, on page 72

- ip ips name, on page 74
- ip ips notify, on page 76
- ip ips sdf location, on page 77
- ip ips signature, on page 79
- ip ips signature-category, on page 81
- ip ips signature-definition, on page 82
- ip ips signature disable, on page 83
- ip kerberos source-interface, on page 84
- ip msdp border, on page 85
- ip mtu, on page 87
- ip nhrp cache non-authoritative, on page 89
- ip nhrp nhs, on page 90
- ip port-map, on page 93
- ip radius source-interface, on page 99
- ip reflexive-list timeout, on page 101
- ip route (vasi), on page 103
- ip scp password, on page 104
- ip scp server enable, on page 105
- ip scp username, on page 107
- ip sdee, on page 108
- ip sdee events, on page 110
- ip security add, on page 111
- ip security aeso, on page 113
- ip security dedicated, on page 115
- ip security eso-info, on page 118
- ip security eso-max, on page 119
- ip security eso-min, on page 121
- ip security extended-allowed, on page 123
- ip security first, on page 125
- ip security ignore-authorities, on page 127
- ip security ignore-cipso, on page 129
- ip security implicit-labelling, on page 131
- ip security multilevel, on page 133
- ip security reserved-allowed, on page 135
- ip security strip, on page 137
- ip sftp password, on page 139
- ip sftp username, on page 140

ip inspect

To apply a set of inspection rules to an interface, use the ip inspect command in interface configuration mode. There are two different modes for this command, configuration mode and interface configuration mode. To remove the set of rules from the interface, use the **no** form of this command.

Global Configuation Mode
ip inspect *inspection-name* {**in** | **out**} [**redundancy** | **stateful hsrp-group-name** | **update seconds***seconds*]
no ip inspect *inspection-name* {**in** | **out**} [**redundancy** | **stateful hsrp-group-name** | **update seconds***seconds*]

Interface Configuration Mode
ip inspect *inspection-name* {**in** | **out**} [**redundancy** | **stateful hsrp-group-name**]
no ip inspect *inspection-name* {**in** | **out**} [**redundancy** | **stateful hsrp-group-name**]

Syntax Description

Interface Configuration Mode	
inspection-name	Identifies which set of inspection rules to apply.
in	Applies the inspection rules to inbound interface.
out	Applies the inspection rules to outbound interface.
redunancy	Enables reduncany.
stateful	Enables stateful redundancy.
hsrp-group-name	The hsrp-group name that is used to configure box-to-box HA
Global Configuration Mode	Redundancy settings for firewall sessions
redundancy	
update	Update settings for firewall HA sessions
seconds <*10-60*>	The time interval between consecutive updates. The default is 10 seconds.

Command Default

If no set of inspection rules is applied to an interface, no traffic will be inspected by CBAC. If **redundancy stateful <hsrp-grp-name>** is not used, there will be no stateful firewall high-availability.

Command Modes

Interface configuration mode(conf-if)

Command History

Release	**Modification**
11.2	This command was introduced.
12.4(6)T	Added support for **redunancy, update**, **seconds**, and **stateful** keywords.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(33)SRA.

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

Use this command to apply a set of inspection rules to an interface.

Typically, if the interface connects to the external network, you apply the inspection rules to outbound traffic; alternately, if the interface connects to the internal network, you apply the inspection rules to inbound traffic.

In the Interface Configuration mode, use **ip inspect<name> in/out redundancy stateful <hsrp-group>**command. Use the redundancy stateful <hsrp-grp> option to turn on stateful high availability for all session that come up on this inspect rule. The incoming IP traffic is the return traffic of an existing session. It not necessary to have redundancy stateful HSRP group name if you do not require IOS Firewall High availability.

In the Global Configuration mode, use **ip inspect redundancy update seconds <10-60>**. Use the redundancy update seconds option to configure the time interval between the synchronization of the active and standby firewall HA sessions.

Examples

The following example applies a set of inspection rules named MY-INSPECT_RULE to serial0 interface's outbound traffic. This causes the inbound IP traffic to be permitted only if the traffic is part of an existing session, and to be denied if the traffic is not part of an existing session.

```
interface serial0
ip inspect MY-INSPECT_RULE out redundancy stateful B2B-HA-HSRP-GRP
```

Related Commands

Command	Description
ip inspect name	Defines a set of inspection rules.

ip inspect alert-off

To disable Context-based Access Control (CBAC) alert messages, which are displayed on the console, use the **ip inspect alert-off**command in global configuration mode. To enable CBAC alert messages, use the **no** form of this command.

ip inspect alert-off [**vrf** *vrf-name*]
no ip inspect alert-off [**vrf** *vrf-name*]

Syntax Description

vrf *vrf-name*	(Optional) Disables CBAC alert messages only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

Alert messages are displayed.

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 disables CBAC alert messages:

```
ip inspect alert-off
```

ip inspect audit-trail

To turn on Context-based Access Control (CBAC) audit trail messages, which will be displayed on the console after each CBAC session closes, use the **ip inspect audit-trail** command in global configuration mode. To turn off CBAC audit trail messages, use the **no** form of this command.

ip inspect audit-trail [**vrf** *vrf-name*]
no ip inspect audit-trail [**vrf** *vrf-name*]

Syntax Description

vrf *vrf-name*	(Optional) Turns on CBAC audit trail messages only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

Audit trail messages are not displayed.

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

Use this command to turn on CBAC audit trail messages.

Examples

The following example turns on CBAC audit trail messages:

```
ip inspect audit-trail
```

Afterward, audit trail messages such as the following are displayed. These messages are examples of audit trail messages. To determine which protocol was inspected, see the port number of the responder. The port number follows the IP address of the responder.

```
%FW-6-SESS_AUDIT_TRAIL: tcp session initiator (192.168.1.13:33192) sent 22 bytes -- responder
 (192.168.129.11:25) sent 208 bytes
%FW-6-SESS_AUDIT_TRAIL: ftp session initiator 192.168.1.13:33194) sent 336 bytes -- responder
 (192.168.129.11:21) sent 325 bytes
```

The following example disables CBAC audit trail messages for VRF interface vrf1:

```
no ip inspect audit-trail vrf vrf1
```

Following are examples of audit trail messages:

```
00:10:15: %FW-6-SESS_AUDIT_TRAIL: VRF-vrf1:Stop udp session: initiator (192.168.14.1:40801)
```

```
 sent 54 bytes -- responder (192.168.114.1:7) sent 54 bytes
00:10:47: %FW-6-SESS_AUDIT_TRAIL: VRF-vrf1:Stop ftp-data session: initiator (192.168.114.1:20)
 sent 80000 bytes -- responder (192.168.14.1:38766) sent 0 bytes
00:10:47: %FW-6-SESS_AUDIT_TRAIL: VRF-vrf1:Stop ftp session: initiator (192.168.14.1:38765)
 sent 80 bytes -- responder (192.168.114.1:21) sent 265 bytes
00:10:57: %FW-6-SESS_AUDIT_TRAIL: VRF-vrf1:Stop rcmd session: initiator (192.168.14.1:531)
 sent 31 bytes -- responder (192.168.114.1:514) sent 12 bytes
00:10:57: %FW-6-SESS_AUDIT_TRAIL: VRF-vrf1:Stop rcmd-data session: initiator
(192.168.114.1:594) sent 0 bytes -- responder (192.168.14.1:530) sent 0 bytes
```

ip inspect dns-timeout

To specify the Domain Name System (DNS) idle timeout (the length of time during which a DNS name lookup session will still be managed while there is no activity), use the **ip inspect dns-timeout**command in global configuration mode. To reset the timeout to the default of 5 seconds, use the **no** form of this command.

ip inspect dns-timeout *seconds* [**vrf** *vrf-name*]
no ip inspect dns-timeout *seconds* [**vrf** *vrf-name*]

Syntax Description

seconds	Specifies the length of time in seconds, for which a DNS name lookup session will still be managed while there is no activity. The default is 5 seconds.
vrf *vrf-name*	(Optional) Specifies the DNS idle timeout only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

5 seconds

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

When the software detects a valid User Datagram Protocol (UDP) packet for a new DNS name lookup session, if Context-based Access Control (CBAC) inspection is configured for UDP, the software establishes state information for the new DNS session.

If the software detects no packets for the DNS session for a time period defined by the DNS idle timeout, the software will not continue to manage state information for the session.

The DNS idle timeout applies to all DNS name lookup sessions inspected by CBAC.

The DNS idle timeout value overrides the global UDP timeout. The DNS idle timeout value also enters aggressive mode and overrides any timeouts specified for specific interfaces when you define a set of inspection rules with the **ip inspect name** command.

Examples

The following example sets the DNS idle timeout to 30 seconds:

```
ip inspect dns-timeout 30
```

The following example sets the DNS idle timeout back to the default (5 seconds):

```
no ip inspect dns-timeout
```

ip inspect hashtable

To change the size of the session hash table, use the **ip inspect hashtable** command in global configuration mode. To restore the size of the session hash table to the default, use the **no** form of this command.

ip inspect hashtable *number*
no ip inspect hashtable *number*

Syntax Description

number	Size of the hash table in terms of buckets. Possible values for the hash table are 1024, 2048, 4096, and 8192; the default value is 1024.

Command Default

1024 buckets

Command Modes

Global configuration

Command History

Release	Modification
12.2(8)T	This command was introduced.

Usage Guidelines

Use the **ip inspect hashtable** command to increase the size of the hash table when the number of concurrent sessions increases or to reduce the search time for the session. Collisions in a hash table result in poor hash function distribution because many entries are hashed into the same bucket for certain patterns of addresses. Even if a hash function distribution evenly dispenses the input across all of the buckets, a small hash table size will not scale well if there are a large number of sessions. As the number of sessions increase, the collisions increase, which increases the length of the linked lists, thereby, deteriorating the throughput performance.

Note You should increase the hash table size when the total number of sessions running through the context-based access control (CBAC) router is approximately twice the current hash size; decrease the hash table size when the total number of sessions is reduced to approximately half the current hash size. Essentially, try to maintain a 1:1 ratio between the number of sessions and the size of the hash table.

Examples

The following example shows how to change the size of the session hash table to 2048 buckets:

```
ip inspect hashtable 2048
```

ip inspect L2-transparent dhcp-passthrough

To allow a transparent firewall to forward Dynamic Host Control Protocol (DHCP) pass-through traffic, use the **ip inspect L2-transparent dhcp-passthrough**command in global configuration mode. To return to the default functionality, use the **no** form of this command.

ip inspect L2-transparent dhcp-passthrough
no ip inspect L2-transparent dhcp-passthrough

Syntax Description

This command has no arguments or keywords.

Command Default

This command is not enabled; thus, DHCP packets are forwarded or denied according to the configured access control list (ACL).

Command Modes

Global configuration

Command History

Release	Modification
12.3(7)T	This command was introduced.

Usage Guidelines

A transparent firewall allows a Cisco IOS Firewall (a Layer 3 device) to operate as a Layer 2 firewall in bridging mode. Thus, the firewall can exist "transparently" to a network, no longer requiring users to reconfigure their statically defined network devices.

The **ip inspect L2-transparent dhcp-passthrough** command overrides the ACL for DHCP packets; that is, DHCP packets are forwarded even if the ACL is configured to deny all IP packets. Thus, this command can be used to enable a transparent firewall to forward DHCP packets across the bridge without inspection so clients on one side of the bridge can get an IP address from a DHCP server on the opposite side of the bridge.

Examples

Allowing DHCP Pass-Through Traffic

In this example, the static IP address of the client is removed, and the address is acquired via DHCP using the **ip address dhcp** command on the interface that is connected to the transparent firewall.

```
Router# show debug
ARP:
  ARP packet debugging is on
L2 Inspection:
  INSPECT L2 firewall debugging is on
  INSPECT L2 firewall DHCP debugging is on
Router#
Router#
! Configure DHCP passthrough
Router(config)# ip insp L2-transparent dhcp-passthrough
! The DHCP discover broadcast packet arrives from the client. Since this packet is a !
broadcast (255.255.255.255), it arrives in the flood path
*Mar  1 00:35:01.299:L2FW:insp_l2_flood:input is Ethernet0 output is Ethernet1
*Mar  1 00:35:01.299:L2FW*:Src 0.0.0.0 dst 255.255.255.255 protocol udp
*Mar  1 00:35:01.299:L2FW:udp ports src 68 dst 67
*Mar  1 00:35:01.299:L2FW:src 0.0.0.0 dst 255.255.255.255
```

```
! The DHCP pass through flag is checked and the packet is allowed
*Mar  1 00:35:01.299:L2FW:DHCP packet seen. Pass-through flag allows the packet
! The packet is a broadcast packet and therefore not sent to CBAC
*Mar  1 00:35:01.299:L2FW*:Packet is broadcast or multicast.PASS
! The DHCP server 97.0.0.23 responds to the client's request
*Mar  1 00:35:01.303:L2FW:insp_l2_flood:input is Ethernet1 output is Ethernet0
*Mar  1 00:35:01.303:L2FW*:Src 172.16.0.23 dst 255.255.255.255 protocol udp
*Mar  1 00:35:01.307:L2FW:udp ports src 67 dst 68
*Mar  1 00:35:01.307:L2FW:src 172.16.0.23 dst 255.255.255.255
*Mar  1 00:35:01.307:L2FW:DHCP packet seen. Pass-through flag allows the packet
*Mar  1 00:35:01.307:L2FW*:Packet is broadcast or multicast.PASS
*Mar  1 00:35:01.311:L2FW:insp_l2_flood:input is Ethernet0 output is Ethernet1
*Mar  1 00:35:01.311:L2FW*:Src 0.0.0.0 dst 255.255.255.255 protocol udp
*Mar  1 00:35:01.311:L2FW:udp ports src 68 dst 67
*Mar  1 00:35:01.311:L2FW:src 0.0.0.0 dst 255.255.255.255
*Mar  1 00:35:01.315:L2FW:DHCP packet seen. Pass-through flag allows the packet
*Mar  1 00:35:01.315:L2FW*:Packet is broadcast or multicast.PASS
*Mar  1 00:35:01.315:L2FW:insp_l2_flood:input is Ethernet1 output is Ethernet0
*Mar  1 00:35:01.323:L2FW*:Src 172.16.0.23 dst 255.255.255.255 protocol udp
*Mar  1 00:35:01.323:L2FW:udp ports src 67 dst 68
*Mar  1 00:35:01.323:L2FW:src 172.16.0.23 dst 255.255.255.255
*Mar  1 00:35:01.323:L2FW:DHCP packet seen. Pass-through flag allows the packet
*Mar  1 00:35:01.323:L2FW*:Packet is broadcast or multicast.PASS
! The client has an IP address (172.16.0.5) and has issued a G-ARP to let everyone know
it's address
*Mar  1 00:35:01.327:IP ARP:rcvd rep src 172.16.0.5 0008.a3b6.b603, dst 172.16.0.5 BVI1
Router#
```

Denying DHCP Pass-Through Traffic

In this example, DHCP pass-through traffic is not allowed (via the **no ip inspect L2-transparent dhcp-passthrough**command). The client is denied when it attempts to acquire a DHCP address from the server.

```
! Deny DHCP pass-through traffic
Router(config)# no ip inspect L2-transparent dhcp-passthrough

! The DHCP discover broadcast packet arrives from the client
*Mar  1 00:36:40.003:L2FW:insp_l2_flood:input is Ethernet0 output is Ethernet1
*Mar  1 00:36:40.003:L2FW*:Src 0.0.0.0 dst 255.255.255.255 protocol udp
*Mar  1 00:36:40.003:L2FW:udp ports src 68 dst 67
*Mar  1 00:36:40.007:L2FW:src 0.0.0.0 dst 255.255.255.255
! The pass-through flag is checked
*Mar  1 00:36:40.007:L2FW:DHCP packet seen. Pass-through flag denies the packet
! The packet is dropped because the flag does not allow DHCP passthrough traffic. Thus, !
the client cannot acquire an address, and it times out
*Mar  1 00:36:40.007:L2FW:FLOOD Dropping the packet after ACL check.
```

Related Commands

Command	Description
debug ip inspect L2-transparent	Enables debugging messages for transparent firewall events.
show ip inspect	Displays Cisco IOS Firewall configuration and session information.

ip inspect log drop-pkt

To log all packets dropped by the firewall, use the **ip inspect log drop-pkt** command in global configuration mode. To return to the default state, use the **no** form of this command.

ip inspect log drop-pkt
no ip inspect log drop-pkt

Syntax Description

This command has no arguments or keywords.

Command Default

Packets dropped by the firewall are not logged.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(7)T1	This command was introduced.
12.3(8)T	This command was integrated into Release 12.3(8)T.

Usage Guidelines

To see the packets that are dropped by the firewall, the ip inspect log drop-pkt command must be enabled.

Examples

The following example shows how to enable the logging of packets dropped by the firewall:

```
Router> enable
Router# configure terminal
Router(config
)# ip inspect log drop-pkt
```

The following example shows a possible message that can be displayed when packets are dropped:

```
*Sep 9 19:56:28.699: %FW-6-DROP_PKT: Dropping tcp pkt 17.2.2.1:0 => 19.2.2.1:0 with ip ident
 229 due to Invalid Header length
*Sep 9 20:30:47.839: %FW-6-DROP_TCP_PKT: Dropping tcp pkt 17.2.2.1:42829 => 19.2.2.1:80 due
 to SYN pkt with illegal flags -- ip ident 23915 tcpflags 40962 seq.no 3928613134 ack 0
*Sep 10 00:30:24.931: %FW-6-DROP_TCP_PKT: Dropping tcp pkt 17.2.2.1:45771 =>
19.2.2.1:80 due to SYN with data or with PSH/URG flags -- ip ident 55001 tcpflags 40962
seq.no 2232798685 ack 0
*Aug 29 21:57:16.895: %FW-6-DROP_PKT: Dropping tcp pkt 17.2.2.1:51613 => 19.2.2.1:80 due
to Out-Of-Order Segment
```

The table below describes messages that occur when packets are dropped.

Table 1: ip inspect log drop-pkt Messages

Field	Description
Invalid Header length	The datagram is so small that it could not contain the layer 4 TCP, Universal Computer Protocol (UCP), or Internet Control Message Protocol (ICMP) header.

Field	Description
Police rate limiting	Rate limiting is enabled, and the packet in question has exceeded the rate limit.
Session limiting	Session limiting is on, and the session count exceeds the configured session threshold.
Bidirectional traffic disabled	Session is unidirectional and the firewall is seeing packets in the other direction and dropping the session.
SYN with data or with PSH/URG flags	TCP SYN packet is seen with data.
Segment matching no TCP connection	Non-initial TCP segment is received without a valid session.
Invalid Segment	There is an invalid TCP segment.
Invalid Seq#	The packet contains an invalid TCP sequence number.
Invalid Ack (or no Ack)	The packet contains an invalid TCP acknowledgement number.
Invalid Flags	Flags in a TCP segment are invalid.
Invalid Checksum	There is an invalid TCP checksum.
SYN inside current window	A synchronization packet is seen within the window of an already established TCP connection.
RST inside current window	A reset (RST) packet is observed within the window of an already established TCP connection.
Out-Of-Order Segment	The packets in a segment are out of order.
Retransmitted Segment with Invalid Flags	A retransmitted packet was already acknowledged by the receiver.
Stray Segment	A TCP segment is received that should not have been received through the TCP state machine such as a TCP SYN packet being received in the listen state.
Internal Error	The TCP state machine that is maintained by the firewall encounters an internal error.
Invalid Window scale option	The responder on one side of a firewall proposes an illegal window scale option. The window scale option is illegal in this case because the initiating side did not propose the option first.
Invalid TCP options	The options in the TCP header are not TCP protocol compliant.

Related Commands

Command	Description
ip inspect tcp block-non-session	Blocks packets that do not belong to the existing firewall TCP sessions in the inbound and outbound directions.

Command	Description
ip inspect tcp finwait-time	Defines how long a TCP session will still be managed after the firewall detects a FIN-exchange.
ip inspect tcp idle-time	Specifies the TCP idle timeout (the length of time a TCP session will still be managed while there is no activity).
ip inspect tcp max-incomplete host	Specifies threshold and blocking time values for TCP host-specific DoS detection and prevention.
ip inspect tcp reassembly	Sets parameters that define how Cisco IOS Firewall application inspection and Cisco IOS IPS will handle out-of-order TCP packets.
ip inspect tcp synwait-time	Defines how long the software will wait for a TCP session to reach the established state before dropping the session.
ip inspect udp idle-time	Specifies the UDP idle timeout (the length of time for which a UDP session will still be managed while there is no activity).

ip inspect max-incomplete high

To define the number of existing half-open sessions that will cause the software to start deleting half-open sessions, use the **ip inspect max-incomplete high**command in global configuration mode. To reset the threshold to the default of 500 half-open sessions, use the **no** form of this command.

ip inspect max-incomplete high *number* [**vrf** *vrf-name*]
no ip inspect max-incomplete high

Syntax Description

number	Specifies the number of existing half-open sessions that will cause the software to start deleting half-open sessions . The default is 500 half-open sessions.
vrf *vrf-name*	(Optional) Defines the number of existing half-open sessions only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

500 half-open sessions

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 unusually high number of half-open sessions (either absolute or measured as the arrival rate) could indicate that a denial-of-service attack is occurring. For TCP, “half-open” means that the session has not reached the established state. For User Datagram Protocol (UDP), “half-open” means that the firewall has detected traffic from one direction only.

Context-based Access Control (CBAC) measures both the total number of existing half-open sessions and the rate of session establishment attempts. Both TCP and UDP half-open sessions are counted in the total number and rate measurements. Measurements are made once a minute.

When the number of existing half-open sessions rises above a threshold (the **max-incomplete high** number), the software will delete half-open sessions as required to accommodate new connection requests. The software will continue to delete half-open requests as necessary, until the number of existing half-open sessions drops below another threshold (the **max-incomplete low** number).

The global value specified for this threshold applies to all TCP and UDP connections inspected by CBAC.

Examples

The following example causes the software to start deleting half-open sessions when the number of existing half-open sessions rises above 900, and to stop deleting half-open sessions when the number drops below 800:

```
ip inspect max-incomplete high 900
ip inspect max-incomplete low 800
```

The following example shows an ALERT_ON message generated for the **ip inspect max-incomplete high** command:

```
ip inspect max-incomplete high 20 vrf vrf1
show log / include ALERT_ON
00:59:00:%FW-4-ALERT_ON: VRF-vrf1:getting aggressive, count (21/20) current 1-min rate: 21
```

Related Commands

Command	Description
ip inspect max-incomplete low	Defines the number of existing half-open sessions that will cause the software to stop deleting half-open sessions.
ip inspect one-minute high	Defines the rate of new unestablished sessions that will cause the software to start deleting half-open sessions.
ip inspect one-minute low	Defines the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions.
ip inspect tcp max-incomplete host	Specifies the threshold and blocking time values for TCP host-specific DoS detection and prevention.

ip inspect max-incomplete low

To define the number of existing half-open sessions that will cause the software to stop deleting half-open sessions, use the **ip inspect max-incomplete low**command in global configuration mode. To reset the threshold to the default of 400 half-open sessions, use the **no** form of this command.

ip inspect max-incomplete low *number* [**vrf** *vrf-name*]
no ip inspect max-incomplete low

Syntax Description

number	Specifies the number of existing half-open sessions that will cause the software to stop deleting half-open sessions . The default is 400 half-open sessions.
vrf *vrf-name*	(Optional) Defines the number of existing half-open sessions only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

400 half-open sessions

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 unusually high number of half-open sessions (either absolute or measured as the arrival rate) could indicate that a denial-of-service attack is occurring. For TCP, “half-open” means that the session has not reached the established state. For User Datagram Protocol (UDP), “half-open” means that the firewall has detected traffic from one direction only.

Context-based Access Control (CBAC) measures both the total number of existing half-open sessions and the rate of session establishment attempts. Both TCP and UDP half-open sessions are counted in the total number and rate measurements. Measurements are made once a minute.

When the number of existing half-open sessions rises above a threshold (the **max-incomplete high** number), the software will delete half-open sessions as required to accommodate new connection requests. The software will continue to delete half-open requests as necessary, until the number of existing half-open sessions drops below another threshold (the **max-incomplete low** number).

The global value specified for this threshold applies to all TCP and UDP connections inspected by CBAC.

Examples

The following example causes the software to start deleting half-open sessions when the number of existing half-open sessions rises above 900, and to stop deleting half-open sessions when the number drops below 800:

```
ip inspect max-incomplete high 900
ip inspect max-incomplete low 800
```

The following example shows an ALERT_OFF message generated for the **ip inspect max-incomplete low**command:

```
ip inspect max-incomplete low 10 vrf vrf1
show log / include ALERT_OFF
00:59:31: %FW-4-ALERT_OFF: VRF-vrf1:calming down, count (9/10) current 1-min rate: 100
```

Related Commands

Command	Description
ip inspect max-incomplete high	Defines the number of existing half-open sessions that will cause the software to start deleting half-open sessions.
ip inspect one-minute high	Defines the rate of new unestablished sessions that will cause the software to start deleting half-open sessions.
ip inspect one-minute low	Defines the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions.
ip inspect tcp max-incomplete host	Specifies the threshold and blocking time values for TCP host-specific DoS detection and prevention.

ip inspect name

To define a set of inspection rules, use the **ip inspect name** command in global configuration mode. To remove the inspection rule for a protocol or to remove the entire set of inspection rules, use the **no** form of this command.

ip inspect name *inspection-name* [**parameter max-sessions** *number*] *protocol* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]
no ip inspect name *inspection-name* [**parameter max-sessions** *number*] *protocol* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]

HTTP Inspection Syntax
ip inspect name *inspection-name* **http** [**java-list** *access-list*] [**urlfilter**] [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]
no ip inspect name *inspection-name protocol*

Simple Mail Transfer Protocol (SMTP) and Extended SMTP Inspection (ESMTP) Syntax
ip inspect name *inspection-name* {**smtp** | **esmtp**} [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**max-data** *number*] [**timeout** *seconds*]

remote-procedure call (RPC) Inspection Syntax
ip inspect name *inspection-name* [**parameter max-sessions** *number*] **rpc program-number** *number* [**wait-time** *minutes*] [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]
no ip inspect name *inspection-name protocol*

Post Office Protocol 3(POP3)/ Internet Message Access Protocol(IMAP) Inspection Syntax
ip inspect name *inspection-name* **imap** [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**reset**] [**secure-login**] [**timeout** *number*]
ip inspect name *inspection-name* **pop3** [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**reset**] [**secure-login**] [**timeout** *number*]

Fragment Inspection Syntax
ip inspect name *inspection-name* [**parameter max-sessions** *number*] **fragment** [**max** *number* **timeout** *seconds*]
no ip inspect name *inspection-name* [**parameter max-sessions** *number*] **fragment** [**max** *number* **timeout** *seconds*]

Application Firewall Provisioning Syntax
ip inspect name *inspection-name* [**parameter max-sessions** *number*] **appfw** *policy-name*
no ip inspect name *inspection-name* [**parameter max-sessions** *number*] **appfw** *policy-name*

User-Defined Application Syntax
ip inspect *inspection-name user-10* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]
noip inspect *inspection-name user-10* [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**timeout** *seconds*]

Session Limiting Syntax
no ip inspect name *inspection-name* [**parameter max-sessions** *number*]

Syntax Description

inspection-name	N ame the set of inspection rules. If you want to add a protocol to an existing set of rules, use the same *inspection-name* as the existing set of rules. **Note** The *inspection-name* cannot exceed 16 characters; otherwise, the name will be truncated to the 16-character limit.
parameter max-sessions *number*	(Optional) Limits the number of established firewall sessions that a firewall rule creates. By default, there is no limit to the number of firewall sessions.
protocol	A protocol keyword listed in the tables below.
alert {**on** | **off**}	(Optional) For each inspected protocol, the generation of alert messages can be set be **on** or **off**. If no option is selected, alerts are generated on the basis of the setting of the **ip inspect alert-off** command.
audit-trail {**on** | **off**}	(Optional) For each inspected protocol, **audit trail**can be set **on** or **off**. If no option is selected, an audit trail message is generated depending on the configuration of the **ip inspect audit-trail**command.
timeout *seconds*	(Optional) To override the global TCP or UDP, or Internet Control Message Protocol (ICMP) idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP, UDP, or ICMP timeouts but will not override the global Domain Name System (DNS) timeout.
http	Specifies the HTTP protocol for Java applet blocking.
java-list *access-list*	(Optional) Specifies the numbered standard access list to use to determine "friendly" sites. This keyword is available only for the HTTP protocol, for Java applet blocking. Java blocking works only with numbered standard access lists.
urlfilter	(Optional) Associates URL filtering with HTTP inspection.
smtp | **esmtp**	Specifies the protocol being used to inspect the traffic.
max-data *number*	(Optional) Specifies the maximum amount of data, in bytes,that can be transferred in a single Simple Mail Transport Protocol (SMTP) session. After the maximum value is exceeded, the firewall logs an alert message and closes the session. The default value is 20MB.
rpc program-number *number*	Specifies the program number to permit. This keyword is available only for the remote-procedure call (RPC) protocol.
wait-time *minutes*	(Optional) Specifies the number of minutes to keep a small gap in the firewall to allow subsequent connections from the same source address and to the same destination address and port. The default wait-time is zero minutes. This keyword is available only for the RPC protocol.
imap	Specifies that the Internet Message Access Protocol (IMAP) is being used.
reset	(Optional) Resets the TCP connection if the client enters a nonprotocol command before authentication is complete.

secure-login	(Optional) Causes a user at a nonsecure location to use encryption for authentication.
pop3	Specifies that the Post Office Protocol, Version 3 (POP3) is being used.
fragment	Specifies fragment inspection for the named rule.
max *number*	(Optional) Specifies the maximum number of unassembled packets for which state information (structures) is allocated by Cisco IOS software. Unassembled packets are packets that arrive at the router interface before the initial packet for a session. The acceptable range is 50 through 10000. The default is 256 state entries. • Memory is allocated for the state structures, and setting this value to a larger number may cause memory resources to be exhausted.
timeout *seconds* (fragmentation)	(Optional) Configures the number of seconds that a packet state structure remains active. When the timeout value expires, the router drops the unassembled packet, freeing that structure for use by another packet. The default timeout value is 1 second. • If this number is set to a value greater that 1 second, it is automatically adjusted by the Cisco IOS software when the number of free state structures goes below certain thresholds: when the number of free states is fewer than 32, the timeout is divided by 2. When the number of free states is fewer than 16, the timeout is set to 1 second.
appfw	Specifies application firewall provisioning.
policy-name	Application firewall policy name. **Note** This name must match the name specified via the **appfw policy-name** command.

Command Default

No inspection rules are defined.

Command Modes

Global configuration (config)

Command History

Release	Modification
11.2P	This command was introduced.
12.0(5)T	This command was modified. Support was added for configurable alert and audit trail, IP fragmentation checking, and NetShow protocol.
12.2(11)YU	This command was modified. Support was added for ICMP and Session Initiation Protocol (SIP) protocols. The **urlfilter** keyword was added to the HTTP inspection syntax.
12.2(15)T	This command was modified. Support was added for ICMP, SIP, and the **urlfilter** keyword was added.
12.3(1)	This command was modified. Skinny protocol support was added.

Release	Modification
12.3(7)T	This command was modified. Extended Simple Mail Transfer Protocol (ESMTP) protocol support was added.
12.3(14)T	This command was modified.The **appfw** keyword and the *policy-name* argument were added to support application firewall provisioning. The **parameter max-sessions**, **reset**, **router-traffic,**and **secure-login**, and keywordswere added. Support for a larger list of protocols including user-defined applications was added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.(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.
15.1(1)T	This command was modified. Support for the CU-SeeMe protocol and the **cuseeme** keyword was removed.

Usage Guidelines

To define a set of inspection rules, enter the **ip inspect name**command for each protocol that you want the Cisco IOS firewall to inspect, using the same *inspection-name*. Give each set of inspection rules a unique *inspection-name*, which should not exceed the 16-character length limit. Define either one or two sets of rules per interface--you can define one set to examine both inbound and outbound traffic, or you can define two sets: one for outbound traffic and one for inbound traffic. The **no ip inspect-name** *protocol* removes the inspection rule for the specified protocol.

no ip inspect name command removes the entire set of inspection rules.

To define a single set of inspection rules, configure inspection for all the desired application-layer protocols, and for ICMP, TCP, and UDP, or as desired. This combination of TCP, UDP, and application-layer protocols join together to form a single set of inspection rules with a unique name. (There are no application-layer protocols associated with ICMP.)

To remove the inspection rule for a protocol, use the **no** form of this command with the specified inspection name and protocol; To remove the entire set of inspection rules, use the **no** form of this command only; that is, do not list any inspection names or protocols.

In general, when inspection is configured for a protocol, return traffic entering the internal network will be permitted only if the packets are part of a valid, existing session for which state information is being maintained.

Table 2: Protocol Keywords--Transport-Layer and Network-Layer Protocols

Protocol	Keyword
ICMP	icmp
TCP	**tcp**
UDP	**udp**

Note The TCP, UDP, and H.323 protocols support the **router-traffic** keyword, which enables inspection of traffic destined to or originated from a router. The command format is as follows: **ip inspect name** *inspection-name* {**tcp** | **udp** | **H323**} [**alert** {**on** | **off**}] [**audit-trail** {**on** | **off**}] [**router-traffic**][**timeout** *seconds*]

TCP and UDP Inspection

You can configure TCP and UDP inspection to permit TCP and UDP packets to enter the internal network through the firewall, even if the application-layer protocol is not configured to be inspected. However, TCP and UDP inspection do not recognize application-specific commands, and therefore might not permit all return packets for an application, particularly if the return packets have a different port number from the previous exiting packet.

Any application-layer protocol that is inspected will take precedence over the TCP or UDP packet inspection. For example, if inspection is configured for FTP, all control channel information will be recorded in the state table, and all FTP traffic will be permitted back through the firewall if the control channel information is valid for the state of the FTP session. The fact that TCP inspection is configured is irrelevant.

With TCP and UDP inspection, packets entering the network must exactly match an existing session. The entering packets must have the same source or destination addresses and source or destination port numbers as the exiting packet (but reversed). Otherwise, the entering packets will be blocked at the interface.

Granular protocol inspection allows you to specify TCP or UDP ports by using the port-to-application mapping (PAM) table. This eliminates having to inspect all applications running under TCP or UDP and the need for multiple ACLs to filter the traffic.

Using the PAM table, you can pick an existing application or define a new one for inspection, thereby simplifying Access Control List (ACL) configuration.

ICMP Inspection

ICMP inspection sessions are done on the basis of the source address of the inside host that originates the ICMP packet. Dynamic ACLs are created for return ICMP packets of the allowed types (echo-reply,destination unreachable, time-exceeded, and timestamp reply) for each session. No port numbers associated with an ICMP session, and the permitted IP address of the return packet is a wild-card in the ACL. The wildcard address is because the IP address of the return packet cannot be known in advance for time-exceeded and destination-unreachable replies. These replies can come from intermediate devices rather than the intended destination.

Application-Layer Protocol Inspection

In general, if you configure inspection for an application-layer protocol, packets for that protocol should be permitted to exit the firewall (by configuring the correct ACL), and packets for that protocol will be allowed back in through the firewall only if they belong to a valid existing session. Each protocol packet is inspected to maintain information about the session state.

Java, H.323, RPC, SIP, and SMTP inspection have additional information, described in the next five sections. The table below lists the supported application-layer protocols.

Table 3: Protocol Keywords--Application-Layer Protocols

Protocol	Keyword
Application Firewall	**appfw**
CU-SeeMe	**cuseeme**

Protocol	Keyword
ESMTP	**smtp**
FTP	**ftp**
IMAP	**imap**
Java	**http**
H.323	**h323**
Microsoft NetShow	**netshow**
POP3	**pop3**
RealAudio	**realaudio**
RPC	**rpc**
SIP	**sip**
Simple Mail Transfer Protocol (SMTP)	**smtp**
Skinny Client Control Protocol (SCCP)	**skinny**
StreamWorks	**streamworks**
Structured Query Language*Net (SQL*Net)	**sqlnet**
TFTP	**tftp**
UNIX R commands (rlogin, rexec, rsh)	**rcmd**
VDOLive	**vdolive**
WORD	**user-defined application name** ; **use prefix -user** **Note** All applications that appear under the **show ip port-map** command are supported.

Java Inspection

Java inspection enables Java applet filtering at the firewall. Java applet filtering distinguishes between trusted and untrusted applets by relying on a list of external sites that you designate as "friendly." If an applet is from a friendly site, the firewall allows the applet through. If the applet is not from a friendly site, the applet will be blocked. Alternately, you could permit applets from all sites except sites specifically designated as "hostile."

Note Before you configure Java inspection, you must configure a numbered standard access list that defines "friendly" and "hostile" external sites. You configure this numbered standard access list to permit traffic from friendly sites, and to deny traffic from hostile sites. If you do not configure a numbered standard access list, but use a "placeholder" access list in the **ip inspect name** *inspection-name* **http** command, all Java applets will be blocked.

Note Java blocking forces a strict order on TCP packets. To properly verify that Java applets are not in the response, a firewall will drop any TCP packet that is out of order. Because the network--not the firewall--determines how packets are routed, the firewall cannot control the order of the packets; the firewall can only drop and retransmit all TCP packets that are not in order.

Caution Context-Based Access Control (CBAC) does not detect or block encapsulated Java applets. Therefore, Java applets that are wrapped or encapsulated, such as applets in .zip or .jar format, are *not* blocked at the firewall. CBAC also does not detect or block applets loaded via FTP, gopher, or HTTP on a nonstandard port.

H.323 Inspection

If you want CBAC inspection to work with NetMeeting 2.0 traffic (an H.323 application-layer protocol), you must also configure inspection for TCP, as described in the chapter "Configuring Context-Based Access Control" in the *Cisco IOS Security Configuration Guide* . This requirement exists because NetMeeting 2.0 uses an additional TCP channel not defined in the H.323 specification.

RPC Inspection

RPC inspection allows the specification of various program numbers. You can define multiple program numbers by creating multiple entries for RPC inspection, each with a different program number. If a program number is specified, all traffic for that program number will be permitted. If a program number is not specified, all traffic for that program number will be blocked. For example, if you created an RPC entry with the NFS program number, all NFS traffic will be allowed through the firewall.

SIP Inspection

You can configure SIP inspection to permit media sessions associated with SIP-signaled calls to traverse the firewall. Because SIP is frequently used to signal both incoming and outgoing calls, it is often necessary to configure SIP inspection in both directions on a firewall (both from the protected internal network and from the external network). Because inspection of traffic from the external network is not done with most protocols, it may be necessary to create an additional inspection rule to cause only SIP inspection to be performed on traffic coming from the external network.

SMTP Inspection

SMTP inspection causes SMTP commands to be inspected for illegal commands. Packets with illegal commands are modified to a "xxxx" pattern and forwarded to the server. This process causes the server to send a negative reply, forcing the client to issue a valid command. An illegal SMTP command is any command except the following:

- DATA
- HELO

- HELP
- MAIL
- NOOP
- QUIT
- RCPT
- RSET
- SAML
- SEND
- SOML
- VRFY

ESMTP Inspection

Like SMTP, ESMTP inspection also causes the commands to be inspected for illegal commands. Packets with illegal commands are modified to a "xxxx" pattern and forwarded to the server. This process causes the server to send a negative reply, forcing the client to issue a valid command. An illegal ESMTP command is any command except the following:

- AUTH
- DATA
- EHLO
- ETRN
- HELO
- HELP
- MAIL
- NOOP
- QUIT
- RCPT
- RSET
- SAML
- SEND
- SOML
- VRFY

In addition to inspecting commands, the ESMTP firewall also inspects the following extensions via deeper command inspection:

- Message Size Declaration (SIZE)

- Remote Queue Processing Declaration (ETRN)
- Binary MIME (BINARYMIME)
- Command Pipelining
- Authentication
- Delivery Status Notification (DSN)
- Enhanced Status Code (ENHANCEDSTATUSCODE)
- 8bit-MIMEtransport (8BITMIME)

Note SMTP and ESMTP cannot exist simultaneously. An attempt to configure both protocols will result in an error message.

Use of the urlfilter Keyword

If you specify the **urlfilter** keyword, the Cisco IOS Firewall will interact with a URL filtering software to control web traffic for a given host or user on the basis of a specified security policy.

Note Enabling HTTP inspection with or without any option triggers the Java applet scanner, which is CPU intensive. The only way to stop the Java applet scanner is to specify the **java-list** *access-list* option. Configuring URL filtering without enabling the **java-list** *access-list*option will severely impact performance.

Use of the timeout Keyword

If you specify a timeout for any of the transport-layer or application-layer protocols, the timeout will override the global idle timeout for the interface to which the set of inspection rules is applied.

If the protocol is TCP or a TCP application-layer protocol, the timeout will override the global TCP idle timeout. If the protocol is UDP or a UDP application-layer protocol, the timeout will override the global UDP idle timeout.

If you do not specify a timeout for a protocol, the timeout value applied to a new session of that protocol will be taken from the corresponding TCP or UDP global timeout value valid at the time of session creation.

The default ICMP timeout is deliberately short (10 seconds) due to the security hole that is opened by allowing ICMP packets with a wild-card source address back into the inside network. The timeout will occur 10 seconds after the last outgoing packet from the originating host. For example, if you send a set of 10 ping packets spaced one second apart, the timeout will expire in 20 seconds or 10 seconds after the last outgoing packet. However, the timeout is not extended for return packets. If a return packet is not seen within the timeout window, the gap will be closed and the return packet will not be allowed in. Although the default timeout can be made longer if desired, it is recommended that this value be kept relatively short.

IP Fragmentation Inspection

CBAC inspection rules can help protect hosts against certain denial-of-service attacks involving fragmented IP packets. Even though the firewall keeps an attacker from making actual connections to a given host, the attacker may still be able to disrupt services provided by that host. This is done by sending many noninitial IP fragments or by sending complete fragmented packets through a router with an ACL that filters the first

fragment of a fragmented packet. These fragments can tie up resources on the target host as it tries to reassemble the incomplete packets.

Using fragmentation inspection, the firewall maintains an *interfragment state* (structure) for IP traffic. Noninitial fragments are discarded unless the corresponding initial fragment was permitted to pass through the firewall. Noninitial fragments received before the corresponding initial fragments are discarded.

Note Fragmentation inspection can have undesirable effects in certain cases, because it can result in the firewall discarding any packet whose fragments arrive out of order. There are many circumstances that can cause out-of-order delivery of legitimate fragments. Apply fragmentation inspection in situations where legitimate fragments, which are likely to arrive out of order, might have a severe performance impact.

Because routers running Cisco IOS software are used in a very large variety of networks, and because the CBAC feature is often used to isolate parts of internal networks from one another, the fragmentation inspection feature is not enabled by default. Fragmentation detection must be explicitly enabled for an inspection rule using the **ip inspect name** command. Unfragmented traffic is never discarded because it lacks a fragment state. Even when the system is under heavy attack with fragmented packets, legitimate fragmented traffic, if any, will still get some fraction of the firewall's fragment state resources, and legitimate, unfragmented traffic can flow through the firewall unimpeded.

Application Firewall Provisioning

Application firewall provisioning allows you to configure your Cisco IOS Firewall to detect and prohibit a specific protocol type of traffic.

Most firewalls provide packet filtering capabilities that simply permit or deny traffic without inspecting the data stream; the Cisco IOS application firewall can detect whether a packet is in compliance with a given HTTP protocol. If the packet is determined to be unauthorized, it will be dropped, the connection will be reset, and a syslog message will be generated, as appropriate.

User-Defined Applications

You can define your own applications and enter them into the PAM table using the **ip port-map** command. Then you set up your inspection rules by inserting your user-defined application as a value for the *protocol* argument in the **ip inspect name** command.

Session Limiting

Users can limit the number of established firewall sessions that a firewall rule creates by setting the "max-sessions" threshold. A session counter is maintained for each firewall interface. When a session count exceeds the specified threshold, an alert FW-4-SESSION_THRESHOLD_EXCEEDED message is logged to the syslog server and no new sessions can be created.

Examples

The following example causes the software to inspect TCP sessions and UDP sessions, and to specifically allow CU-SeeMe, FTP, and RPC traffic back through the firewall for existing sessions only. For UDP traffic, audit-trail is on. For FTP traffic, the idle timeout is set to override the global TCP idle timeout. For RPC traffic, program numbers 100003, 100005, and 100021 are permitted.

```
ip inspect name myrules tcp
ip inspect name myrules udp audit-trail on
ip inspect name myrules cuseeme
ip inspect name myrules ftp timeout 120
ip inspect name myrules rpc program-number 100003
```

```
ip inspect name myrules rpc program-number 100005
ip inspect name myrules rpc program-number 100021
```

The following example adds fragment checking to software inspection of TCP and UDP sessions for the rule named *"myrules*." In this example, the firewall software will allocate 100 state structures, and the timeout value for dropping unassembled packets is set to 4 seconds. If 100 initial fragments for 100 different packets are sent through the router, all of the state structures will be used up. The initial fragment for packet 101 will be dropped. Additionally, if the number of free state structures (structures available for use by unassembled packets) drops below the threshold values, 32 or 16, the timeout value is automatically reduced to 2 or 1, respectively. Changing the timeout value frees up packet state structures more quickly.

```
ip inspect name myrules tcp
ip inspect name myrules udp audit-trail on
ip inspect name myrules cuseeme
ip inspect name myrules ftp timeout 120
ip inspect name myrules rpc program-number 100003
ip inspect name myrules rpc program-number 100005
ip inspect name myrules rpc program-number 100021
ip inspect name myrules fragment max 100 timeout 4
```

The following firewall and SIP example shows how to allow outside-initiated calls and internal calls. For outside-initiated calls, an ACL needs to be accessed to allow for the traffic from the initial signaling packet from outside. Subsequent signaling and media channels will be allowed by the inspection module.

```
ip inspect name voip sip
interface FastEthernet0/0
 ip inspect voip in
!
!
interface FastEthernet0/1
 ip inspect voip in
 ip access-group 100 in
!
!
access-list 100 permit udp host <gw ip> any eq 5060
access-list 100 permit udp host <proxy ip> any eq 5060
access-list deny ip any any
```

The following example shows two configured inspections named fw_only and fw_urlf; URL filtering will work only on the traffic that is inspected by fw_urlf. Note that the **java-list** *access-list*option has been enabled, which disables java scanning.

```
ip inspect name fw_only http java-list 51 timeout 30
interface e0
 ip inspect fw_only in
!
ip inspect name fw_urlf  http java-list 51 urlfilter timeout 30
interface e1
 ip inspect fw_urlf in
```

The following example shows how to define the HTTP application firewall policy mypolicy. This policy includes all supported HTTP policy rules. This example also includes sample output from the **show appfw configuration** and **show ip inspect config** commands, which allow you to verify the configured setting for the application policy.

```
! Define the HTTP policy.
```

```
appfw policy-name mypolicy
 application http
  strict-http action allow alarm
  content-length maximum 1 action allow alarm
  content-type-verification match-req-rsp action allow alarm
  max-header-length request 1 response 1 action allow alarm
  max-uri-length 1 action allow alarm
  port-misuse default action allow alarm
  request-method rfc default action allow alarm
  request-method extension default action allow alarm
  transfer-encoding type default action allow alarm
!
!
! Apply the policy to an inspection rule.
ip inspect name firewall appfw mypolicy
ip inspect name firewall http
!
!
! Apply the inspection rule to all HTTP traffic entering the FastEthernet0/0 interface.
interface FastEthernet0/0
 ip inspect firewall in
!
!
! Issue the show appfw configuration
 command and the show ip inspect config
command after the inspection rule "mypolicy" is applied to all incoming HTTP traffic on the
 FastEthernet0/0 interface.
!
Router# show appfw configuration

Application Firewall Rule configuration
  Application Policy name mypolicy
    Application http
      strict-http action allow alarm
      content-length minimum 0 maximum 1 action allow alarm
      content-type-verification match-req-rsp action allow alarm
      max-header-length request length 1 response length 1 action allow alarm
      max-uri-length 1 action allow alarm
      port-misuse default action allow alarm
      request-method rfc default action allow alarm
      request-method extension default action allow alarm
      transfer-encoding default action allow alarm
Router# show ip inspect config

Session audit trail is disabled
Session alert is enabled
one-minute (sampling period) thresholds are [400:500] connections
max-incomplete sessions thresholds are [400:500]
max-incomplete tcp connections per host is 50. Block-time 0 minute.
tcp synwait-time is 30 sec -- tcp finwait-time is 5 sec
tcp idle-time is 3600 sec -- udp idle-time is 30 sec
dns-timeout is 5 sec
Inspection Rule Configuration
Inspection name firewall
http alert is on audit-trail is off timeout 3600
```

Related Commands

Command	Description
ip inspect	Applies a set of inspection rules to an interface.
ip inspect alert-off	Disables CBAC alert messages.

Command	Description
ip inspect audit trail	Turns on CBAC audit trail messages, which will be displayed on the console after each CBAC session close.

ip inspect one-minute high

To define the rate of new unestablished sessions that will cause the software to start deleting half-open sessions, use the **ip inspect one-minute high**command in global configuration mode. To reset the threshold to the default of 500 half-open sessions, use the **no** form of this command.

ip inspect one-minute high *number* [**vrf** *vrf-name*]
no ip inspect one-minute high

Syntax Description

number	Specifies the rate of new unestablished TCP sessions that will cause the software to start deleting half-open sessions . The default is 500 half-open sessions.
vrf *vrf-name*	(Optional) Defines the information only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

500 half-open sessions

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 unusually high number of half-open sessions (either absolute or measured as the arrival rate) could indicate that a denial-of-service attack is occurring. For TCP, “half-open” means that the session has not reached the established state. For User Datagram Protocol (UDP), “half-open” means that the firewall has detected traffic from one direction only.

Context-based Access Control (CBAC) measures both the total number of existing half-open sessions and the rate of session establishment attempts. Both TCP and UDP half-open sessions are included in the total number and rate measurements. Measurements are made once a minute.

When the rate of new connection attempts rises above a threshold (the **one-minute high** number), the software will delete half-open sessions as required to accommodate new connection attempts. The software will continue to delete half-open sessions as necessary, until the rate of new connection attempts drops below another threshold (the **one-minute low** number). The rate thresholds are measured as the number of new session connection attempts detected in the last one-minute sample period. (The rate is calculated as an exponentially decayed rate.)

The global value specified for this threshold applies to all TCP and UDP connections inspected by CBAC.

Examples

The following example causes the software to start deleting half-open sessions when more than 1000 session establishment attempts have been detected in the last minute, and to stop deleting half-open sessions when fewer than 950 session establishment attempts have been detected in the last minute:

```
ip inspect one-minute high 1000
ip inspect one-minute low 950
```

Related Commands

Command	Description
ip inspect one-minute low	Defines the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions.
ip inspect max-incomplete high	Defines the number of existing half-open sessions that will cause the software to start deleting half-open sessions.
ip inspect max-incomplete low	Defines the number of existing half-open sessions that will cause the software to stop deleting half-open sessions.
ip inspect tcp max-incomplete host	Specifies the threshold and blocking time values for TCP host-specific DoS detection and prevention.

ip inspect one-minute low

To define the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions, use the **ip inspect one-minute low**command in global configuration mode. To reset the threshold to the default of 400 half-open sessions, use the **no** form of this command.

ip inspect one-minute low *number* [**vrf** *vrf-name*]
no ip inspect one-minute low

Syntax Description

number	Specifies the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions . The default is 400 half-open sessions.
vrf *vrf-name*	(Optional) Defines the information only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

400 half-open sessions

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 unusually high number of half-open sessions (either absolute or measured as the arrival rate) could indicate that a denial-of-service attack is occurring. For TCP, “half-open” means that the session has not reached the established state. For User Datagram Protocol (UDP), “half-open” means that the firewall has detected traffic from one direction only.

Context-based Access Control (CBAC) measures both the total number of existing half-open sessions and the rate of session establishment attempts. Both TCP and UDP half-open sessions are included in the total number and rate measurements. Measurements are made once a minute.

When the rate of new connection attempts rises above a threshold (the **one-minute high** number), the software will delete half-open sessions as required to accommodate new connection attempts. The software will continue to delete half-open sessions as necessary, until the rate of new connection attempts drops below another threshold (the **one-minute low** number). The rate thresholds are measured as the number of new session connection attempts detected in the last one-minute sample period. (The rate is calculated as an exponentially decayed rate.)

The global value specified for this threshold applies to all TCP and UDP connections inspected by CBAC.

Examples

The following example causes the software to start deleting half-open sessions when more than 1000 session establishment attempts have been detected in the last minute, and to stop deleting half-open sessions when fewer than 950 session establishment attempts have been detected in the last minute:

```
ip inspect one-minute high 1000
ip inspect one-minute low 950
```

Related Commands

Command	Description
ip inspect max-incomplete high	Defines the number of existing half-open sessions that will cause the software to start deleting half-open sessions.
ip inspect max-incomplete low	Defines the number of existing half-open sessions that will cause the software to stop deleting half-open sessions.
ip inspect one-minute high	Defines the rate of new unestablished sessions that will cause the software to start deleting half-open sessions.
ip inspect tcp max-incomplete host	Specifies the threshold and blocking time values for TCP host-specific DoS detection and prevention.

ip inspect tcp block-non-session

To block packets that do not belong to the existing firewall TCP sessions in the inbound and outbound directions, use the **ip inspect tcp block-non-session** command in global configuration mode. To return to the default state, use the **no** form of this command.

ip inspect tcp block-non-session [**vrf vrf-name**]
no inspect tcp block-non-session [**vrf vrf-name**]

Syntax Description

vrf	(Optional) Declares a specific VPN routing/forwarding instance (VRF).
vrf-name	(Optional) Name of the VRF.

Command Default

TCP packets that do not belong to an existing TCP session on the firewall are allowed through the firewall.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(6)	This command was introduced.
12.3(7)T	This command was integrated into Release 12.3(6)T.
12.3(7)XI	This command was integrated into the Release 12.3(7)XI.
12.3(14)T	The vrf keyword and vrf-name argument were added.

Usage Guidelines

This command will deny TCP packets that do not belong to an existing TCP session the firewall knows about. To be applicable, the following conditions must be met:

- The TCP packets should traverse interfaces where a firewall rule is applicable.
- The TCP packets should be non-connection initiating (that is, packets without the SYN bit set in them). For connection initiating packets, the existing rules of session creation would apply.

Examples

The following example shows how to configure the firewall to block any externally initiated TCP sessions:

```
Router> enable
Router# config terminal
Router(config
)# ip inspect tcp block-non-session
```

Related Commands

Command	Description
ip inspect log drop-pkt	Logs all packets dropped by the firewall.

Command	Description
ip inspect tcp finwait-time	Defines how long a TCP session will still be managed after the firewall detects a FIN-exchange.
ip inspect tcp idle-time	Specifies the TCP idle timeout (the length of time a TCP session will still be managed while there is no activity).
ip inspect tcp max-incomplete host	Specifies threshold and blocking time values for TCP host-specific (DoS) detection and prevention.
ip inspect tcp reassembly	Sets parameters that define how Cisco IOS Firewall application inspection and Cisco IOS IPS will handle out-of-order TCP packets.
ip inspect tcp synwait-time	Defines how long the software will wait for a TCP session to reach the established state before dropping the session.
ip inspect udp idle-time	Specifies the UDP idle timeout (the length of time for which a UDP session will still be managed while there is no activity).

ip inspect tcp finwait-time

To define how long a TCP session will be managed after the firewall detects a finish (FIN)-exchange, use the **ip inspect tcp finwait-time** command in global configuration mode. To reset the timeout to the default of 5 seconds, use the **no** form of this command.

ip inspect tcp finwait-time *seconds* [**vrf** *vrf-name*]
no ip inspect tcp finwait-time

Syntax Description

seconds	Specifies how long a TCP session will be managed after the firewall detects a FIN-exchange. The default is 5 seconds. Valid values are from 1 to 2147483. If the FIN-exchange completes within the configured finwait time, the connection is closed normally.
vrf *vrf-name*	(Optional) Defines the information only for the specified VPN routing and forwarding (VRF) interface.

Command Default

The default management time is 5 seconds.

Command Modes

Global configuration (config)

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	This command was modified. The **vrf** *vrf-name* keyword and argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.(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

When the software detects a valid TCP packet that is the first in a session, and if Context-Based Access Control (CBAC) inspection is configured for the protocol of the packet, the software establishes state information for the new session.

Use this command to define how long a TCP session state information will be maintained after the firewall detects a FIN-exchange for the session. The FIN-exchange occurs when the TCP session is ready to close. In a TCP connection, the client and the server terminate their end of the connection by sending a FIN message. The time that the client and the server wait for their FIN message to be acknowledged by each other before closing the sequence during a TCP connection is called the finwait time. The timeout that you set for the finwait time is referred to as the finwait timeout.

The global value specified for the finwait timeout applies to all TCP sessions inspected by CBAC.

Examples

The following example shows how to change the finwait timeout to 10 seconds:

```
ip inspect tcp finwait-time 10
```

The following example shows how to change the finwait timeout back to the default (5 seconds):

```
no ip inspect tcp finwait-time
```

Related Commands

Command	Description
show ip inspect	Displays CBAC configuration and session information.

ip inspect tcp idle-time

To specify the TCP idle timeout (the length of time a TCP session will still be managed while there is no activity), use the **ip inspect tcp idle-time**command in global configuration mode. To reset the timeout to the default of 3600 seconds (1 hour), use the **no** form of this command.

ip inspect tcp idle-time *seconds* [**vrf** *vrf-name*]
no ip inspect tcp idle-time

Syntax Description

seconds	Specifies the length of time, in seconds, for which a TCP session will still be managed while there is no activity. The default is 3600 seconds (1 hour).
vrf *vrf-name*	(Optional) Specifies the TCP idle timer only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

3600 seconds (1 hour)

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

When the software detects a valid TCP packet that is the first in a session, and if Context-based Access Control (CBAC) inspection is configured for the packet's protocol, the software establishes state information for the new session.

If the software detects no packets for the session for a time period defined by the TCP idle timeout, the software will not continue to manage state information for the session.

The global value specified for this timeout applies to all TCP sessions inspected by CBAC. This global value can be overridden for specific interfaces when you define a set of inspection rules with the **ip inspect name**(global configuration) command.

Note This command does not affect any of the currently defined inspection rules that have explicitly defined timeouts. Sessions created based on these rules still inherit the explicitly defined timeout value. If you change the TCP idle timeout with this command, the new timeout will apply to any new inspection rules you define or to any existing inspection rules that do not have an explicitly defined timeout. That is, new sessions based on these rules (having no explicitly defined timeout) will inherit the global timeout value.

Examples

The following example sets the global TCP idle timeout to 1800 seconds (30 minutes):

```
ip inspect tcp idle-time 1800
```

The following example sets the global TCP idle timeout back to the default of 3600 seconds (one hour):

```
no ip inspect tcp idle-time
```

ip inspect tcp max-incomplete host

To specify threshold and blocking time values for TCP host-specific denial-of-service (DoS) detection and prevention, use the **ip inspect tcp max-incomplete host**command in global configuration mode. To reset the threshold and blocking time to the default values, use the **no** form of this command.

ip inspect tcp max-incomplete host *number* **block-time** *minutes* [**vrf** *vrf-name*]
no ip inspect tcp max-incomplete host

Syntax Description

number	Specifies how many half-open TCP sessions with the same host destination address can exist at a time, before the software starts deleting half-open sessions to the host. Use a number from 1 to 250. The default is 50 half-open sessions.
block-time	Specifies blocking of connection initiation to a host.
minutes	Specifies how long the software will continue to delete new connection requests to the host. The default is 0 minutes.
vrf *vrf-name*	(Optional) Specifies the information only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

50 half-open sessions and 0 minutes

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 unusually high number of half-open sessions with the same destination host address could indicate that a denial-of-service attack is being launched against the host. For TCP, “half-open” means that the session has not reached the established state.

Whenever the number of half-open sessions with the same destination host address rises above a threshold (the **max-incomplete host** number), the software will delete half-open sessions according to one of the following methods:

- If the **block-time** *minutes* timeout is 0 (the default):

The software will delete the oldest existing half-open session for the host for every new connection request to the host. This ensures that the number of half-open sessions to a given host will never exceed the threshold.

- If the **block-time** *minutes* timeout is greater than 0:

The software will delete all existing half-open sessions for the host, and then block all new connection requests to the host. The software will continue to block all new connection requests until the **block-time** expires.

The software also sends syslog messages whenever the **max-incomplete host** number is exceeded and when blocking of connection initiations to a host starts or ends.

The global values specified for the threshold and blocking time apply to all TCP connections inspected by Context-based Access Control (CBAC).

Examples

The following example changes the max-incomplete host number to 40 half-open sessions, and changes the block-time timeout to 2 minutes:

```
ip inspect tcp max-incomplete host 40 block-time 2
```

The following example resets the defaults (50 half-open sessions and 0 minutes):

```
no ip inspect tcp max-incomplete host
```

Related Commands

Command	Description
ip inspect max-incomplete high	Defines the number of existing half-open sessions that will cause the software to start deleting half-open sessions.
ip inspect max-incomplete low	Defines the number of existing half-open sessions that will cause the software to stop deleting half-open sessions.
ip inspect one-minute high	Defines the rate of new unestablished sessions that will cause the software to start deleting half-open sessions.
ip inspect one-minute low	Defines the rate of new unestablished TCP sessions that will cause the software to stop deleting half-open sessions.

ip inspect tcp reassembly

To set parameters that define how Cisco IOS Firewall application inspection and Cisco IOS Intrusion Prevention System (IPS) will handle out-of-order TCP packets, use the **ip inspect tcp reassembly** command in global configuration mode. To disable at least one defined parameter, use the **no** form of this command.

ip inspect tcp reassembly {**alarm** {**on** | **off**} | **memory limit** *size-in-kb* | **queue length** *number-of-packets* | **timeout** *seconds*} [**vrf** *vrf-name*]
no ip inspect tcp reassembly {**alarm** | **queue length** | **timeout** | **memory limit**} [**vrf** *vrf-name*]

Syntax Description

alarm {**on** | **off**	Specifies the alert message configuration. If enabled, a syslog message is generated when an out-of-order packet is dropped. Default value: **on**
memory	Specifies the memory use allowed by the TCP reassembly module.
limit *size-in-kb*	Specifies the limit of out of order queue size.
queue	Specifies the out of order queue parameters.
length *number-of-packets*	Maximum number of out-of-order packets that can be held per queue (buffer). (There are two queues per session.) Available value range: 0 to 1024. Default value: 16. **Note** If the queue length is set to 0, all out-of-order packets are dropped; that is, TCP out-of-order packet buffering and reassembly is disabled.
timeout *seconds*	Number of seconds the TCP reassembly module will hold out-of-order segments that are waiting for the first segment missing in the sequence. After the timeout timer has expired, a retry timer is started. The value for the retry timer is four times the configured timeout value.
vrf *vrf-name*	Specifies the VPN routing and forwarding (VRF) parameter and name.

Command Default

Queue length: 16

Memory Limit: 1024 kilobytes

Alarm: on

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(11)T	This command was introduced.
15.0(1)M	This command was integrated into Cisco IOS Release 15.0(1)M.

Usage Guidelines

The queue length Value

The value specified for the queue length is applicable for two queues per session: one queue is for the initiator traffic and the other queue is for the responder traffic. For example, the default queue size is 16. Thus, up to 16 packets can be held per queue, so 16 packets per queue results in a maximum of 32 packets per session.

When the maximum queue length value is reached, the packet being switched is dropped unless it is the packet that will be processed by a firewall or IPS. If the packet is dropped, a syslog message, which explains why the packet was dropped, will be generated. (To generate syslog messages, you must have the alarm option set to "on.")

The timeout Value

When a timer expires for the first time, the packets in the queue are not deleted. However, after the retry timer expires, the session is deleted, a syslog message is generated, and all unprocessed, out-of-order packets still in the queue are deleted.

The memory limit Value

When the limit for TCP reassembly memory is reached, packets from the reassembly queue of the current session are released so incoming packets can be accepted. Packets from the end of the queue are released to ensure that they are farthest away from the hole that is to be filled. However, if the queue is empty and the maximum memory has been reached, the incoming packet is dropped.

The alarm Value

If an alarm value is not configured, the value is set to "on," unless the **ip inspect alarm** command is enabled and set to off; thus, syslog messages related to TCP connections will not be generated. However, if the alarm value for this command is set to "on" and the **ip inspect alarm** command is set to "off," the value of the **ip inspect alarm** command is ignored and syslog messages are generated.

The alarm value is independent of and in addition to the syslog messages that can be enabled for a Cisco IOS Firewall or Cisco IOS IPS.

Examples

The following example shows how to instruct Cisco IOS IPS how to handle out-of-order packets for TCP connections:

```
Router(config)#
ip inspect tcp reassembly queue length 18
Router(config)#
ip inspect tcp reassembly memory limit 200
```

Related Commands

Command	Description
ip inspect tcp block-non-session	Blocks packets that do not belong to the existing firewall TCP sessions in the inbound and outbound directions.

ip inspect tcp synwait-time

To define how long the software will wait for a TCP session to reach the established state before dropping the session, use the **ip inspect tcp synwait-time**command in global configuration mode. To reset the timeout to the default of 30 seconds, use the **no** form of this command.

ip inspect tcp synwait-time *seconds* [**vrf** *vrf-name*]
no ip inspect tcp synwait-time

Syntax Description

seconds	Specifies how long, in seconds, the software will wait for a TCP session to reach the established state before dropping the session . The default is 30 seconds.
vrf *vrf-name*	(Optional) Defines the information only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

30 seconds

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

Use this command to define how long Cisco IOS software will wait for a TCP session to reach the established state before dropping the session. The session is considered to have reached the established state after the first synchronize sequence number (SYN) bit of the session is detected.

The global value specified for this timeout applies to all TCP sessions inspected by Context-based Access Control (CBAC).

Examples

The following example changes the synwait timeout to 20 seconds:

```
ip inspect tcp synwait-time 20
```

The following example changes the synwait timeout back to the default (30 seconds):

```
no ip inspect tcp synwait-time
```

ip inspect tcp window-scale-enforcement loose

To configure Cisco IOS software to disable the window scale option check for a TCP packet that has an invalid window scale option under the Context-Based Access Control (CBAC) firewall, use the **ip inspect tcp window-scale-enforcement loose** command in global configuration mode. To return to the command default, use the **no** form of this command.

ip inspect tcp window-scale-enforcement loose
no ip inspect tcp window-scale-enforcement loose

Command Default

The strict window scale option check is enabled in the firewall by default.

Command Modes

Global configuration (config)

Command History

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

Usage Guidelines

The window scale extension expands the definition of the TCP window to 32 bits and then uses a scale factor to carry this 32-bit value in the 16-bit Window field of the TCP header. Cisco IOS software enforces strict checking of the TCP window scale option. See section 2 of RFC1323, "TCP Window Scale Option," for more information on this function.

There are occasions when a server may be using a non-RFC compliant TCP/IP protocol stack. In this case, the initiator does not offer the window scale option, but the responder has the option enabled with a window scale factor that is not zero.

Cisco IOS administrators who experience issues with a noncompliant server may not have control over the client to which they need to connect. Disabling the Cisco IOS firewall to connect to the noncompliant server is not desirable and may fail if each endpoint cannot agree on the window scaling factor to use for its respective receive window.

The **ip inspect tcp window-scale-enforcement loose** command is used in global configuration mode to allow noncompliant window scale negotiation and works without the firewall being disabled to access the noncompliant servers. This command works under the CBAC firewall, which intelligently filters TCP and UDP packets based on application-layer protocol session information. CBAC inspects traffic that travels through the firewall to discover and manage state information for TCP and UDP sessions. CBAC is configured using an inspect rule only on interfaces. This state information is used to create temporary openings in the firewall's access lists to allow return traffic and additional data connections for permissible sessions. Traffic entering or leaving the configured interface is inspected based on the direction that the inspect rule was applied.

Examples

The following example configures the IOS to disable the window scale option check in the CBAC firewall for a TCP packet that has an invalid window scale option:

```
Router# config
Router(config)# ip inspect tcp window-scale-enforcement loose
```

Related Commands

Command	Description
ip inspect tcp synwait-time	Configures the length of time the software waits for a TCP session to reach the established state before dropping the session.

ip inspect udp idle-time

To specify the User Datagram Protocol (UDP) idle timeout (the length of time for which a UDP "session" will still be managed while there is no activity), use the **ip inspect udp idle-time** command in global configuration mode. To reset the timeout to the default of 30 seconds, use the **no** form of this command.

ip inspect udp idle-time *seconds* [**vrf** *vrf-name*]
no ip inspect udp idle-time

Syntax Description

seconds	Specifies the length of time a UDP "session" will still be managed while there is no activity . The default is 30 seconds.
vrf *vrf-name*	(Optional) Specifies the UDP idle timeout only for the specified Virtual Routing and Forwarding (VRF) interface.

Command Default

30 seconds

Command Modes

Global configuration

Command History

Release	Modification
11.2 P	This command was introduced.
12.3(14)T	The **vrf** *vrf-name*keyword/argument pair was added.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

When the software detects a valid UDP packet, if Context-based Access Control (CBAC) inspection is configured for the packet's protocol, the software establishes state information for a new UDP "session." Because UDP is a connectionless service, there are no actual sessions, so the software approximates sessions by examining the information in the packet and determining if the packet is similar to other UDP packets (for example, it has similar source or destination addresses) and if the packet was detected soon after another similar UDP packet.

If the software detects no UDP packets for the UDP session for the a period of time defined by the UDP idle timeout, the software will not continue to manage state information for the session.

The global value specified for this timeout applies to all UDP sessions inspected by CBAC. This global value can be overridden for specific interfaces when you define a set of inspection rules with the **ip inspect name**command.

Note This command does not affect any of the currently defined inspection rules that have explicitly defined timeouts. Sessions created based on these rules still inherit the explicitly defined timeout value. If you change the UDP idle timeout with this command, the new timeout will apply to any new inspection rules you define or to any existing inspection rules that do not have an explicitly defined timeout. That is, new sessions based on these rules (having no explicitly defined timeout) will inherit the global timeout value.

Examples

The following example sets the global UDP idle timeout to 120 seconds (2 minutes):

```
ip inspect udp idle-time 120
```

The following example sets the global UDP idle timeout back to the default of 30 seconds:

```
no ip inspect udp idle-time
```

ip inspect waas enable

To enable the zone-based firewall to inspect Cisco Wide Area Application Service (WAAS) traffic, use the **ip inspect waas enable** command in global configuration mode. To disable the firewall inspection of WAAS traffic, use the **no** form of this command.

ip inspect waas enable
no ip inspect waas enable

Syntax Description

This command has no arguments or keywords.

Command Default

WAAS traffic inspection is not enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS Release 12.4(11)T2	This command was introduced.

Usage Guidelines

Because the WAAS automatic discovery process uses TCP options before sending UDP traffic, the firewall must be configured to pass TCP options. Use the **ip inspect waas enable** command to configure the firewall to allow TCP options.

Examples

The following example shows how to enable the firewall inspection of WAAS traffic:

```
Device# configure terminal
Device(config)# ip inspect waas enable
```

integrity

To specify one or more integrity algorithms for an Internet Key Exchange Version 2 (IKEv2) proposal, use the **integrity** command in IKEv2 proposal configuration mode. To remove the configuration of the hash algorithm, use the **no** form of this command.

integrity *integrity type*
no integrity

Syntax Description

integrity type	Specifies the hash algorithm.

Command Default

The default integrity algorithm is used.

Command Modes

IKEv2 proposal configuration (config-ikev2-proposal)

Command History

Release	Modification
15.1(1)T	This command was introduced.
15.1(2)T	This command was modified. The **sha256** and **sha384** keywords were added.
Cisco IOS XE Release 3.3S	This command was integrated into Cisco IOS XE Release 3.3S.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.

Usage Guidelines

Note Security threats, as well as the cryptographic technologies to help protect against them, are constantly changing. For more information about the latest Cisco cryptographic recommendations, see the Next Generation Encryption (NGE) white paper.

Use this command to specify the integrity algorithm to be used in an IKEv2 proposal. The default integrity algorithms in the default proposal are SHA-1 and MD5. The integrity algorithms can be one of the following:

Integrity Type	Description
sha1	Specifies Secure Hash Algorithm (SHA-1 - HMAC variant) as the hash algorithm (No longer recommended).
sha256	Specifies SHA-2 family 256-bit (HMAC variant) as the hash algorithm.
sha384	Specifies SHA-2 family 384-bit (HMAC variant) as the hash algorithm.
sha512	Specifies SHA-2 family 512-bit (HMAC variant) as the hash algorithm.

Integrity Type	Description
md5	Specifies Message-Digest algorithm 5 (MD5 - HMAC variant) as the hash algorithm.

Note You cannot selectively remove an integrity algorithm when multiple integrity algorithms are configured.

Suite-B adds support for the SHA-2 family (HMAC variant) hash algorithm used to authenticate packet data and verify the integrity verification mechanisms for the IKEv2 proposal configuration. HMAC is a variant that provides an additional level of hashing.

Examples

The following example configures an IKEv2 proposal with the MD5 integrity algorithm:

```
Device(config)# crypto ikev2 proposal proposal1
Device(config-ikev2-proposal)# integrity md5
```

Related Commands

Command	Description
crypto ikev2 proposal	Defines an IKEv2 proposal.
encryption (ikev2 proposal)	Specifies the encryption algorithm in an IKEv2 proposal.
group (ikev2 proposal)	Specifies the Diffie-Hellman group identifier in an IKEv2 proposal.
show crypto ikev2 proposal	Displays the parameters for each IKEv2 proposal.

ip interface

To configure a virtual gateway IP interface on a Secure Socket Layer Virtual Private Network (SSL VPN) gateway, use the **ip interface** command in webvpn gateway configuration mode. To disable the configuration, use the **no** form of this command.

ip interface *type number* [**port** {**443***port-number*}]
no ip interface

Syntax Description

type	Interface type. For more information, use the question mark (?) online help function.
number	Interface or subinterface number. For more information about the numbering syntax for your networking device, use the question mark (?) online help function.
port	(Optional) Configures a specific port on the gateway.
443	(Optional) Configures the default secure port.
port-number	(Optional) Port number to be configured on the SSL VPN gateway. Range: 1025 to 65535. Default: 443.

Command Default

The command is disabled. The virtual gateway IP address is not configured.

Command Modes

Webvpn gateway configuration (config-webvpn-gateway)

Command History

Release	Modification
12.4(24)T	This command was introduced.

Usage Guidelines

The **ip interface** command is used to configure a interface on a SSL VPN gateway. You can use this command to configure the WebVPN gateway to retrieve the IP address from an interface, and if you do not want to configure the IP address manually. This command is useful when the public interface is Dynamic Host Configuration Protocol (DHCP) and you do not know the IP address or when the IP address gets changed.

If the **ip interface** command is not configured then the WebVPN will use the IP address configured using the **ip address** command.

Examples

The following example shows how to configure a virtual gateway IP interface on port 1036 of an SSL VPN gateway:

```
Router# configure terminal
Router(config)# webvpn gateway gateway1
Router(config-webvpn-gateway)# ip interface FastEthernet 0/1 port 1036
```

Related Commands

Command	Description
ip address	Configures a proxy IP address on an SSL VPN gateway.

Command	Description
webvpn gateway	Defines an SSL VPN gateway and enters WebVPN gateway configuration mode.

ip ips

To apply an Intrusion Prevention System (IPS) rule to an interface, use the **ip ips**command in interface configuration mode. To remove an IPS rule from an interface direction, use the **no** form of this command.

ip ips *ips-name* {**in** | **out**}
no ip ips *ips-name* {**in** | **out**}

Syntax Description

ips-name	Name of IPS signature definition file (SDF).
in	Applies IPS to inbound traffic.
out	Applies IPS to outbound traffic.

Command Default

By default, IPS signatures are not applied to an interface or direction.

Command Modes

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.3(8)T	The command name was changed from the **ip audit**command to the **ip ips**command.

Usage Guidelines

The **ip ips** command loads the SDF onto the router and builds the signature engines when IPS is applied to the first interface.

Note The router prompt disappears while the signatures are loading and the signature engines are building. It will reappear after these tasks are complete. Depending on your platform and how many signatures are being loaded, building the signature engine can take several of minutes. It is recommended that you enable logging messages so you can monitor the engine building status.

The **ip ips** command replaces the **ip audit** command. If the **ip audit**command is part of an existing configuration, IPS will interpret it as the **ip ips**command.

Examples

The following example shows the basic configuration necessary to load the attack-drop.sdf file onto a router running Cisco IOS IPS. Note that the configuration is almost the same as when you load the default signatures onto a router, except for the **ip ips sdf location** command, which specifies the attack-drop.sdf file.

```
!
ip ips sdf location disk2:attack-drop.sdf
ip ips name MYIPS
!
interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
```

```
 ip ips MYIPS in
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
```

The following example shows how to configure the router to load and merge the attack-drop.sdf file with the default signatures. After you have merged the two files, it is recommended to copy the newly merged signatures to a separate file. The router can then be reloaded (via the reload command) or reinitalized to so as to recognize the newly merged file (as shown the following example)

```
!
ip ips name MYIPS
!
interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
 ip ips MYIPS in
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
!
! Merge the flash-based SDF (attack-drop.sdf) with the built-in signatures.
copy disk2:attack-drop.sdf ips-sdf
! Save the newly merged signatures to a separate file.
copy ips-sdf disk2:my-signatures.sdf
!
! Configure the router to use the new file, my-signatures.sdf
configure terminal
ip ips sdf location disk2:my-signatures.sdf
! Reinitialize the IPS by removing the IPS rule set and reapplying the rule set.
interface gig 0/1
 no ip ips MYIPS in
!
*Apr 8 14:05:38.243:%IPS-2-DISABLED:IPS removed from all interfaces - IPS disabled
!
 ip ips MYIPS in
!
 exit
```

Related Commands

Command	Description
copy ips-sdf	Loads or saves the SDF in the router.
ip ips sdf location	Specifies the location in which the router should load the SDF.

ip ips auto-update

To enable automatic signature updates for Cisco IOS Intrusion Prevention System (IPS), use the **ip ips auto-update** command in global configuration mode. To revert back to the default value, use the **no** form of this command.

ip ips auto-update
no ip ips auto-update

Syntax Description

This command has no arguments or keywords.

Command Default

The default value is defined in the signature definition XML.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(11)T	This command was introduced.

Usage Guidelines

Automatic signature updates allow users to override the existing IPS configuration and automatically keep signatures up to date on the basis of a preset time, which can be configured to a preferred setting.

Use the **ip ips auto-update** command to enable Cisco IOS IPS to automatically update the signature file on the system. When enabling automatic signature updates, it is recommended that you ensure the following configuration guidelines have been met:

- The router's clock is set up with the proper relative time.
- The frequency for Cisco IOS IPS to obtain updated signature information has been defined (through the **occur-at** command).
- Automatic signature updates can be enabled from Cisco.com by using the **cisco** command. This command cannot be used in conjunction with the **url** command.
- The URL in which to retrieve the Cisco IOS IPS signature configuration files has been specified (through the **url** command).
- Optionally, the username and password in which to access the files from the server has been specified (through the **username** command). The **username** command would be optional in this case if the username and password command were previously configured through the **ips signature update cisco**command in Priviledged EXEC mode. The user name and password must be configured for updating signatures directly from Cisco.com.

The Default Value

A user or a management station can override the default value through the **category** command or the **signature** command; a value set with either of these commands will be saved as the delta value. The no form of the ip ips auto-update command will remove the delta value and revert back to the default value in the definition XML.

Setting Time for Auto Updates

Cisco IOS time can be updated through the hardware clock or the software configurable clock (which ever option is available on your system). Although Network Time Protocol (NTP) is typically used for automated time synchronization, Cisco IOS IPS updates use the local clock resources as a reference for update intervals. Thus, NTP should be configured to update the local time server of the router, as appropriate.

Examples

The following example shows how to configure automatic signature updates and issue the **show ip ips auto-update** command to verify the configuration. In this example, the signature package file is pulled from the TFTP server at the third hour of the 5 day of the month, at the 56th minute of this hour. (Note that adjustments are made for months without 31 days and daylight savings time.)

```
Router# clock set ?
hh:mm:ss Current Time
Router# clock set 10:38:00 20 apr 2006
Router#
*Apr 20 17:38:00.000: %SYS-6-CLOCKUPDATE: System clock has been updated from 10:37:55 MST
Thu Apr 20 2006 to 10:38:00 MST Thu Apr 20 2006, configured from console by cisco on console.
Router(config)# ip ips auto-update
Router(config-ips-auto-update)# occur-at monthly 5 56 3
Router#
*May 4 2006 15:50:28 MST: IPS Auto Update: setting update timer for next update: 5 days 56
 min 3 hrs
*May 4 2006 15:50:28 MST: %SYS-5-CONFIG_I: Configured from console by cisco on console
Router#
Router# show ip ips auto-update

IPS Auto Update Configuration
URL : tftp://192.168.0.2/jdoe/ips-auto-update/IOS_reqSeq-dw.xml
Username : not configured
Password : not configured
Auto Update Intervals
  minutes (0-59) : 56
  hours (0-23) : 3
  days of month (1-31) : 5
  days of week: (0-6) :
```

Related Commands

Command	Description
occur-at	Defines the frequency in which Cisco IOS IPS obtains updated signature information.
cisco	Enables automatic signature updates from Cisco.com.
url (ips-autoupdate)	Defines a location in which to retrieve the Cisco IOS IPS signature configuration files.
username (ips-autoupdate)	Defines a username and password in which to access signature files from the server.

ip ips config location

To specify the location in which the router will save signature information, use the **ip ips config location**command in global configuration mode. To remove the specified location, use the **no** form of this command.

ip ips config location *url*
no ip ips config location

Syntax Description

url	Location where the signature file is saved. Available URL options: • Local flash, such as flash:sig.xml • FTP server, such as ftp://myuser:mypass@ftp_server.sig.xml • rcp, such as rcp://myuser@rcp_server/sig.xml • TFTP server, such as tftp://tftp_server/sig.xml **Note** If the specified location is a URL, such as an FTP server, the user must have writer privileges.

Command Default

No configuration files are saved.

Command Modes

Global configuration

Command History

Release	Modification
12.4(11)T	This command was introduced.

Usage Guidelines

Before configuring the **ip ips config location**command, you must create a directory for the config location via the **mkdir** command.

The **ip ips config location**command configures a Cisco IOS Intrusion Prevention System (IPS) signature location, which tells Cisco IOS IPS where to save signature information.

The configuration location is used to restore the IPS configuration in cases such as router reboots or IPS becoming disabled or reenabled. Files, such as signature definitions, signature-type definitions, and signature category information, are written in XML format, compressed, and saved to the specified IPS signature location.

Note If a location is not specified, or if a location is removed via the **no** form, no files will be saved.

Note The **ip ips config location** command replaces the **ip ips sdf location**command.

Examples

The following example shows how to instruct the router to save all signature information to the directory “flash:/ips5”:

```
Router# mkdir
flash:/ips5
Create directory filename [ips5]?
Created dir flash:/ips5
Router#
Router#
Router#
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip ips name MYIPS
Router(config)# ip ips config location
flash:/ips5
Router(config)# ip ips signature-category
Router(config-ips-category)# category all
Router(config-ips-category-action)# retired true
Router(config-ips-category-action)# exit
Router(config-ips-category)# category ios_ips advanced
Router(config-ips-category-action)# retired false
Router(config-ips-category-action)# exit

Router(config-ips-category)# exit
Do you want to accept these changes? [confirm]
Router(config)# d
*Nov 14 2006 17:16:42 MST: Applying Category configuration to signatures ..
Router(config)#
```

ip ips deny-action ips-interface

To create an access control list (ACL) filter for the deny actions ("denyFlowInline" and "denyConnectionInline") on the intrusion prevention system (IPS) interface rather than ingress interface, use the **ip ips deny-action ips-interface**command in global configuration mode. To return to the default, use the **no** form of this command.

ip ips deny-action ips-interface
no ip ips deny-action ips-interface

Syntax Description

This command has no arguments or keywords.

Command Default

ACLs filter for the deny actions are applied to the ingress interface.

Command Modes

Global configuration

Command History

Release	Modification
12.3(14)T	This command was introduced.

Usage Guidelines

Use the **ip ips deny-action ips-interface** command to change the default behavior of the ACL filters that are created for the deny actions.

Note You should configure this command only if at least one signature is configured to use the supported deny actions (denyFlowInline and denyConnectionInline, if the input interface is configured to for load balancing, and if IPS is configured on the output interface.

Default ACL Filter Approach

By default, ACL filters for the deny actions are created on the ingress interfaces of the offending packet. Thus, if Cisco IOS IPS is configured in outbound direction on the egress interface and the "deny" ACLs are created on the ingress interface, Cisco IOS IPS will drop the matching traffic before it goes through much processing. Unfortunately, this approach does not work in load balancing scenarios for which there is more than one ingress interface performing load-balancing.

Alternative ACL Filter Approach

T he **ip ips deny-action ips-interface** command enables ACLs to be created on the same interface and in the same direction as Cisco IOS IPS is configured. This alternative approach supports load-balancing scenarios--assuming that the load-balancing interfaces have the same Cisco IOS IPS configuration. However, all outbound Cisco IOS IPS traffic will go through substantial packet path processing before it is eventually dropped by the ACLs.

Examples

The following example shows how to configure load-balancing between interface e0 and interface e1:

```
ip ips name test
```

```
ip ips deny-action ips-interface
! Enables load balancing with e1
interface e0
 ip address 10.1.1.14 255.255.255.0
 no shut
!
! Enables load balancing with e0
interface e1
 ip address 10.1.1.16 255.255.255.0
 no shut
!
interface e2
 ip address 10.1.1.18 255.255.255.0
 ip ips test in
 no shut
```

ip ips enable-clidelta

To enable the signature tuning settings in the clidelta.xmz file on the router to take precedence over the signature settings in the intrusion prevention system (IPS) iosips-sig-delta.xmz file, use the **ip ips enable-clidelta** command in global configuration mode. To restore precedence to the iosips-sig-delta.xmz file settings, use the no form of this command.

ip ips enable-clidelta
no ip ips enable-clidelta

Syntax Description

This command has no arguments or keywords.

Command Default

This command is disabled by default.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.1(2)T	This command was introduced.

Usage Guidelines

Most IPS devices and applications provide either a single default configuration or multiple default configurations. Using one of these default configurations is an ideal starting point for deploying IPS. When IOS IPS is deployed, parameters such as severity, active status, or event actions of certain signatures need to be tuned to meet the requirements of an enterprise network traffic profile.

Once the **ip ips enable-clidelta**commandis enabled, a local cli-delta.xmz file is generated containing the local tuning signatures configured through the CLI. The settings in the clidelta.xmz file take precedence when a globally administered delta signature update, contained in the iosips-sig-delta.xmz file, is sent from a central repository and applied to the configuration of the local router.

Examples

The following example shows how to enable the clidelta functionality:

```
Router(config)# ip ips enable-clidelta
```

Related Commands

Command	Description
show ip ips sig-clidelta	Displays information about the IPS iosips-sig-clidelta.xmz file on the router to verify signature tuning settings.

ip ips event-action-rules

To enter config-rule configuration mode, which allows users to change the target value rating, use the **ip ips event-action-rules** command in global configuration mode.

ip ips event-action-rules

Syntax Description

This command has no arguments or keywords.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
12.4(11)T	This command was introduced.

Usage Guidelines

You must issue the **ip ips event-action-rules** command to define the target value rating via the **target-value** command.

Examples

The following example shows how to change the target value to low for the host 192.168.0.1:

```
configure terminal
ip ips event-action-rules
 target-value low target-address 192.168.0.1
```

Related Commands

Command	Description
target-value	Defines the target value rating for a host.

ip ips fail closed

To instruct the router to drop all packets until the signature engine is built and ready to scan traffic, use the **ip ips fail closed** command in global configuration mode. To return to the default functionality, use the **no** form of this command.

ip ips fail closed
no ip ips fail closed

Syntax Description

This command has no arguments or keywords.

Command Default

All packets are passed without being scanned while the signature engine is being built or if the signature engine fails to build.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

Cisco IOS IPS Fails to Load the SDF

By default, the router running Intrusion Prevention System (IPS) will load the built-in signatures if it fails to load the signature definition file (SDF). If this command is issued, the router will drop all packets--unless the user specifies an access control list (ACL) for packets to send to IPS.

IPS Loads the SDF but Fails to Build a Signature Engine

If the router running IPS loads the SDF but fails to build a signature engine, the router will mark the engine "not ready." If an available engine is previously loaded, the IPS will keep the available engine and discard the engine that is not ready for use. If no previous engines have been loaded or "not ready," the router will install the engine that is not ready and rely on the configuration of the **ip ips fail closed**command.

By default, packets destined for an engine marked "not ready" will be passed without being scanned. If this command is issued, the router will drop all packets that are destined for that signature engine.

Examples

The following example shows how to instruct the router to drop all packets if the SME is not yet available:

```
Router(config)# ip ips fail closed
```

ip ips inherit-obsolete-tunings

Note Effective with Cisco IOS Release 15.2T, the **ip ips inherit-obsolete tunings** command is deprecated because the Cisco IOS IPS Signature Scanning with Lightweight Signatures feature is discontinued.

To enable Cisco IOS Intrusion Prevention System (IPS) signatures to inherit tunings from obsoleted signatures in a Cisco IOS IPS, use the **ip ips inherit-obsolete tunings** command in global configuration mode. To disable this function, use the **no** form of this command.

ip ips inherit-obsolete-tunings
no ip ips inherit-obsolete-tunings

Syntax Description

This command has no arguments or keywords.

Command Default

Tunings from obsoleted signatures in Cisco IOS IPS are not inherited.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced.
15.2T	This command was deprecated.

Usage Guidelines

The ip ips inherit-obsolete-tunings command enables new signatures to obsolete older signatures and inherit the event-action and enabled parameters of the obsolete tuning values without the need to manually tune the new signatures. All other parameter changes, including the "Retire" parameter saved in the old signatures, will be ignored.

After you enter the command, the screen displays a warning message asking you to clarify the intended usage and then asks whether you accept the configuration, By default, old signatures tunings are not inherited by new signatures.

Note The tunings of old signatures will be lost if they are not migrated to new signatures.

Note To enable inheritance of tunings, configure the ip ips inherit-obsolete-tunings command before a signature file is loaded.

Note Users of management devices should use those devices and not enable the ip ips inherit-obsolete-tunings command.

Examples

The following example shows how to configure a router running Cisco IOS IPS to allow new signatures to inherit the tuning values from the obsoleted signatures, without having to manually tune the new signatures:

```
Router(config)# ip ips inherit-obsolete-tunings
```

Related Commands

Command	Description
ip ips	Applies a IPS rule to an interface.
ip ips memory regex chaining	Enables an Cisco IOS IPS to chain multiple regex tables together and load additional signatures.
ip ips memory threshold	Specifies an Cisco IOS IPS memory threshold.
show ip ips	Displays Cisco IOS IPS information such as configured sessions and signatures.

ip ips memory regex chaining

Note Effective with Cisco IOS Release 15.2T, the **ip ips memory regex chaining** command is deprecated because the Cisco IOS IPS Signature Scanning with Lightweight Signatures feature is discontinued.

To enable a Cisco IOS Intrusion Prevention System (IPS) to chain multiple regex tables together and load additional signatures, use the **ip ips memory regex chaining** command in global configuration mode. To disable this function, use the **no** form of this command.

ip ips memory regex chaining
no ip ips memory regex chaining

Syntax Description

This command has no arguments or keywords.

Command Default

Multiple regex table chaining is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced.
15.2T	This command was deprecated.

Usage Guidelines

Multiple regex table chaining is used to load additional signatures when a Cisco IOS IPS is supporting a large signature set. The default is three chained tables when the ip ips memory regex chaining command is enabled. This results in slower performance of Cisco IOS IPS scanning due to scanning packets across more than a single regex table.

When a user tries to load a specific set of signatures that does not fit using a single table, compilation errors will result. A compiler failure error message looks like this:

```
*Sep  9 17:27:46.907: %IPS-4-SIGNATURE_COMPILE_FAILURE: string-tcp 3730:0 - compiles
discontinued for this engine
```

Examples

The following example shows how to enable the ip ips memory regex chaining command:

```
Router(config)# ip ips memory regex chaining
```

Related Commands

Command	Description
ip ips	Applies an IPS rule to an interface.
ip ips inherit-obsolete-tunings	Applies tunings from obsoleted signatures to the new versions of the signatures.

Command	Description
ip ips memory threshold	Specifies a Cisco IOS IPS memory threshold.
show ip ips	Displays Cisco IOS IPS information such as configured sessions and signatures.

ip ips memory threshold

Note Effective with Cisco IOS Release 15.2T, the **ip ips memory threshold** command is deprecated because the Cisco IOS IPS Signature Scanning with Lightweight Signatures feature is discontinued.

To specify a memory threshold when using a Cisco IOS Intrusion Prevention System (IPS), use the **ip ips memory threshold** command in global configuration mode. To disable this function, use the no form of this command.

ip ips memory threshold megabytes
no ip ips memory threshold

Syntax Description

megabytes	The IPS memory threshold, in megabytes. The valid range is from 0-1024.

Command Default

The default IPS memory threshold is 10 percent of free memory--this is available for router operations other than Cisco IOS IPS.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced.
15.2T	This command was deprecated.

Usage Guidelines

The IPS memory threshold defines the amount of free memory unavailable to the IPS.

When you are loading signatures, the default state is that Cisco IOS IPS cannot consume any more memory if the remaining (free) memory becomes less than 10 percent of the size of the total DRAM installed on the router (for example, less than 25.6 MB free memory left on routers with 256 MB DRAM). The 10 percent of free memory unavailable to IPS defines the IPS memory threshold. The IPS memory threshold can be changed using the ip ips memory threshold command to force IPS to use less memory, so that other features get access to more memory if they need it.

Setting a memory threshold for Cisco IOS IPS is recommended especially when an arbitrary number of signatures may be added on top of the recommended sets in Cisco IOS IPS Basic or Advanced/Default categories, or when a fully customized signature set is created and loaded.

Examples

The following example shows how to configure a router running Cisco IOS IPS to set the IPS memory threshold to a value of 50 MB:

```
Router(config)# ip ips memory threshold 50
```

Related Commands

Command	Description
ip ips	Applies an IPS rule to an interface.
ip ips inherit-obsolete-tunings	Applies tunings from obsoleted signatures to the newer versions of the signatures.
ip ips memory regex chaining	Enables a Cisco IOS IPS to chain multiple regex tables together and load additional signatures.
show ip ips	Displays Cisco IOS IPS information such as configured sessions and signatures.

ip ips name

To specify an intrusion prevention system (IPS) rule, use the **ip ips name**command in global configuration mode. To delete an IPS rule, use the **no** form of this command.

ip ips name *ips-name* [**list** *acl*]
no ip ips name *ips-name* [**list** *acl*]

Syntax Description

ips-name	Name for IPS rule.
list *acl*	(Optional) Specifies an extended or standard access control list (ACL) to filter the traffic that will be scanned. **Note** All traffic that is permitted by the ACL is subject to inspection by the IPS. Traffic that is denied by the ACL is not inspected by the IPS.

Command Default

An IPS rule does not exist.

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.3(8)T	The command name was changed from the **ip audit name**command to the **ip ips name**command.

Usage Guidelines

The IPS does not load the signatures until the rule is applied to an interface via the **ip ips**command.

Note This command replaces the **ip audit name** global configuration command. If the **ip audit name** command has been issued in an existing configuration and an access control list (ACL) has been defined, IPS will apply the **ip ips name** command and the ACL parameter on all interfaces that applied the rule.

Examples

The following example shows how to configure a router running Cisco IOS IPS to load the default, built-in signatures. Note that a configuration option for specifying an SDF location is not necessary; built-in signatures reside statically in Cisco IOS.

```
!
ip ips po max-events 100
ip ips name MYIPS
!
interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
 ip ips MYIPS in
 duplex full
 speed 100
```

```
 media-type rj45
 no negotiation auto
!
```

Related Commands

Command	Description
ip ips	Applies an IPS rule to an interface.
show ip ips	Displays IPS information such as configured sessions and signatures.

ip ips notify

To specify the method of event notification, use the **ip ips notify** command in global configuration mode. To disable event notification, use the **no** form of this command.

ip ips notify [**log** | **sdee**]
no ip ips notify [**log** | **sdee**]

Syntax Description

log	(Optional) Send messages in syslog format. **Note** If an option is not specified, alert messages are sent in syslog format.
sdee	(Optional) Send messages in Security Device Event Exchange (SDEE) format.

Command Default

Disabled (alert messages are not sent).

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.3(8)T	The command name was changed from the **ip audit notify**command to the **ip ips notify**command. Also, support for SDEE was introduced, and the **sdee** keyword was added.
12.3(14)T	The Post Office protocol was deprecated, and the **nr-director** keyword was removed.

Usage Guidelines

SDEE is always running, but it does not receive and process events from Intrusion Prevention System (IPS) unless SDEE notification is enabled. If it is not enabled and a client sends a request, SDEE will respond with a fault response message, indicating that notification is not enabled.

To use SDEE, the HTTP server must be enabled (via the **ip http server** command). If the HTTP server is not enabled, the router cannot respond to the SDEE clients because it cannot not see the requests.

Note The **ip ips notify** command replaces the **ip audit notify** command. If the **ip audit notify**command is part of an existing configuration, the IPS will interpret it as the **ip ips notify**command .

Examples

In the following example, event notifications are specified to be sent in SDEE format:

```
ip ips notify sdee
```

Related Commands

Command	Description
ip http server	Enables the HTTP server on your system.

ip ips sdf location

Note In Cisco IOS Release 12.4(11)T, the **ip ips sdf location** command was replaced with the **ip ips config location** command. For more information, see the **ip ips config location** command.

To specify the location in which the router will load the signature definition file (SDF), use the **ip ips sdf location** command in global configuration mode. To remove an SDF location from the configuration, use the **no** form of this command.

ip ips sdf location *url* [**retries** *number* **wait-time** *seconds*] [**autosave**]
no ip ips sdf location *url* [**retries** *number* **wait-time** *seconds*] [**autosave**]

Syntax Description

url	Location of the SDF. Available URL options: • local flash, such as flash:sig.xml • FTP server, such as ftp://myuser:mypass@ftp_server.sig.xml • rcp, such as rcp://myuser@rcp_server/sig.xml • TFTP server, such as tftp://tftp_server/sig.xml
retries *number*	(Optional) Number of times the router will try to load the SDF after the first attempt fails.
wait-time *seconds*	(Optional) Duration, in seconds, between retry attempts.
autosave	(Optional) Specifies that the router will save a new SDF to the specified location.

Command Default

If an SDF location is not specified, the router will load the default built-in signatures.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.
12.4(4)T	The **autosave** keyword was added.
12.4(7.20)T	The **retries** *number* and the **wait-time** *seconds* options were added.
12.4(11)T	This command was replaced with the **ip ips config location** command.

Usage Guidelines

When you specify the **ip ips sdf location** command, the signatures are not loaded until the router is rebooted or until the Intrusion Prevention System (IPS) is applied to an interface (via the **ip ips** command). If IPS is already applied to an interface, the signatures are not loaded. If IPS cannot load the SDF, an error message is issued and the router uses the built-in IPS signatures.

You can also specify the **copy ips-sdf** command to load an SDF from a specified location. Unlike the **ip ips sdf location** command, the signatures are loaded immediately after the **copy ips-sdf** command is entered.

When you specify the **autosave** keyword, the router saves a new SDF to the specified location when signatures are loaded using either the **copy** command or an external management platform such as Security Device Manager (SDM), IPS Management Center (IPSMC) or Cisco Incident Control Server (Cisco ICS). You can specify multiple autosave locations. The router will attempt to save to all autosave locations. The URL must have proper write access permissions.

Examples

The following example shows how to configure the router to load and merge the attack-drop.sdf file with the default signatures. After the files are merged, it is recommended that you copy the merged signatures to a separate file. You can then reload the router (by entering the **reload** command) or reinitialize the router so that it recognizes the newly merged file (as shown the following example).

```
!
ip ips name MYIPS
!
interface GigabitEthernet0/1
 ip address 10.1.1.16 255.255.255.0
 ip ips MYIPS in
 duplex full
 speed 100
 media-type rj45
 no negotiation auto
!
!
! Merge the flash-based SDF (attack-drop.sdf) with the built-in signatures.
copy disk2:attack-drop.sdf ips-sdf
! Save the newly merged signatures to a separate file.
copy ips-sdf disk2:my-signatures.sdf
!
! Configure the router to use the new file, my-signatures.sdf
configure terminal
ip ips sdf location disk2:my-signatures.sdf
! Reinitialize the IPS by removing the IPS rule set and reapplying the rule set.
interface gig 0/1
no ip ips MYIPS in
!
*Apr 8 14:05:38.243:%IPS-2-DISABLED:IPS removed from all interfaces - IPS disabled
!
 ip ips MYIPS in
!
exit
```

Related Commands

Command	Description
copy ips-sdf	Loads or saves the SDF in the router.
ip ips	Applies the IPS rule to an interface.

ip ips signature

Note In Cisco IOS Release 12.4(11)T, the **ip ips signature** command was deprecated.

To attach a policy to a signature, use the **ip ips signature** command in global configuration mode. If the policy disabled a signature, use the **no** form of this command to reenable the signature. If the policy attached an access list to the signature, use the **no** form of this command to remove the access list.

ip ips signature *signature-id* {**delete** | **disable** | **list** *acl-list*}
no ip ips signature *signature-id*

Syntax Description

signature-id	Signature within the signature detection file (SDF).
delete	Deleted a specified signature.
disable	Disables a specified signature.
list *acl-list*	A named, standard, or ACL that is associated with the signature.

Command Default

No policy is attached to a signature.

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.3(8)T	The command name was changed from the **ip audit signature** command to the **ip ips signature** command to support SDFs.
12.4(11)T	This command and support for SDFs were removed.

Usage Guidelines

This command allow you to set three policies: delete a signature, disable the audit of a signature, or qualify the audit of a signature with an access list.

If you are attaching an ACL to a signature, then you also need to create an Intrusion Prevention System (IPS) rule with the **ip ips name**command and apply it to an interface with the **ip ips** command.

Note The **ip ips signature**command replaces the **ip audit signature** command. If the **ip audit signature** command is found in an existing configuration, Cisco IOS IPS will interpret it as the **ip ips signature** command.

Examples

In the following example, a signature is disabled, another signature has ACL 99 attached to it, and ACL 99 is defined:

```
ip ips signature 6150 disable
ip ips signature 1000 list 99
access-list 99 deny 10.1.10.0 0.0.0.255
access-list 99 permit any
```

ip ips signature-category

To enter IPS category (config-ips-category) configuration mode, which allows you to tune Cisco IOS Intrusion Prevention System (IPS) signature parameters on the basis of a signature category, use the **ip ips signature-category** command in global configuration mode.

ip ips signature-category

Syntax Description

This command has no arguments or keywords.

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
12.4(11)T	This command was introduced.

Usage Guidelines

Use the **ip ips signature-category** command if you want to tune signature parameters per category.

Examples

The following example shows how to tune event-action parameters for the signature category "adware/spyware." All tuning information will be applied to all signatures that belong to the adware/spyware category.

```
Router(config)# ip ips signature-category
Router(config-ips-category)# category attack adware/spyware
Router(config-ips-category-action)# event-action produce-alert
Router(config-ips-category-action)# event-action deny-packet-inline
Router(config-ips-category-action)# event-action reset-tcp-connection
Router(config-ips-category-action)# retired false
Router(config-ips-category-action)# ^Z
Do you want to accept these changes? [confirm]y
```

Related Commands

Command	Description
category	Specifies a signature category that is to be used for multiple signature actions or conditions.

ip ips signature-definition

To enter signature-definition-signature configuration mode, which allows you to define a signature for command-line interface (CLI) user tunings, use the **ip ips signature-definition**command in global configuration mode. To revert back to the default value, use the **no** form of this command.

ip ips signature-definition
no ip ips signature-definition

Syntax Description

This command has no arguments or keywords.

Command Default

Signature parameters cannot be defined and default values are used.

Command Modes

Global configuration

Command History

Release	Modification
12.4(11)T	This command was introduced.

Usage Guidelines

Use the **ip ips signature-definition** command to enter signature-definition-signature configuration mode, which allows you to issue the **signature** command. The **signature** command is used to specify a signature whose CLI user tunings are to be customized. After you issue the **signature** command, you can begin to specify which signature parameters (user tunings) are to be changed.

Examples

The following example shows how to modify signature 5081/0 to “produce alert” and “reset tcp connection”:

```
Router(config)# ip ips signature-definition
Router(config-sigdef-sig)# signature 5081 0
Router(config-sigdef-action)# engine
Router(config-sigdef-action-engine)# event-action produce-alert reset-tcp-connection
Router(config-sigdef-action-engine)# ^Z
Do you want to accept these changes:[confirmm]y
```

Related Commands

Command	Description
signature	Specifies a signature for which the CLI user tunings will be changed.

ip ips signature disable

To instruct the router to scan for a given signature but not take any action if the signature is detected , use the **ip ips signature**command in global configuration mode. To reenable a signature, use the **no** form of this command.

ip ips signature *signature-id* [*sub-signature-id*] **disable** [**list** *acl-list*]
no ip ips signature *signature-id* [*sub-signature-id*] **disable** [**list** *acl-list*]

Syntax Description

signature-id *sub-signature-id*	Signature that is disabled.
list *acl-list*	(Optional) A named, standard, or extended access control list (ACL) to filter the traffic that will be scanned. If the packet is permitted by the ACL, the signature will be scanned and reported; if the packet is denied by the ACL, the signature is deemed disabled.

Command Default

All signatures within the signature definition file (SDF) are reported, if detected.

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

You may want to disable a signature (or set of signatures) if your deployment scenario deems the signatures unnecessary.

Examples

The following example shows how to instructs the router not to report on signature 1000, if detected:

```
Router(config) ip ips signature 1000 disable
```

Related Commands

Command	Description
ip ips	Applies the IPS rule to an interface.
ip ips name	Specifies an IPS rule.

ip kerberos source-interface

To specify an interface for the source address of the kerberos packets, use the **ip kerberos source-interface** command in global configuration mode. To disable the configuration, use the **no** form of this command.

ip kerberos source-interface *interface-type number*
no ip kerberos source-interface

Syntax Description

interface-type	Interface type. For more information, use the question mark (?) online help function.
number	Interface or subinterface number. For more information about the numbering syntax for your networking device, use the question mark (?) online help function.

Command Default

An interface for the source address of Kerberos packets is not set.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.
12.2(33)SXI	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SXI.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1 and implemented on the Cisco ASR 1000 Series Aggregation Services Routers.

Examples

The following example shows how to specify an interface for the source address of the Kerberos packets:

```
Router# configure terminal
Router(config)# ip kerberos source-interface FastEthernet 0/0
```

Related Commands

Command	Description
clear kerberos creds	Deletes the contents of the credentials cache.
debug kerberos	Displays information associated with the Kerberos Authentication Subsystem.

ip msdp border

To configure a router that borders a Protocol Independent Multicast (PIM) sparse mode region and dense mode region to use Multicast Source Discovery Protocol (MSDP), use the **ip msdp border**command in global configuration mode. To prevent this action, use the **no** form of this command.

ip msdp [**vrf** *vrf-name*] **border sa-address** *interface-type interface-number*
no ip msdp [**vrf** *vrf-name*] **border sa-address** *interface-type interface-number*

Syntax Description

vrf	(Optional) Supports the multicast VPN routing and forwarding (VRF) instance.
vrf-name	(Optional) Name assigned to the VRF.
sa-address	Specifies the active source IP address.
interface-type interface-number	Interface type and number from which the IP address is derived and used as the rendezvous point (RP) address in Source-Active (SA) messages. Thus, MSDP peers can forward SA messages away from this border. The IP address of the interface is used as the originator ID, which is the RP field in the MSDP SA message. No space is needed between the values.

Command Default

The active sources in the dense mode region will not participate in MSDP.

Command Modes

Global configuration

Command History

Release	Modification
12.0(7)T	This command was introduced.
12.0(23)S	The **vrf** keyword and *vrf-name* argument were added.
12.2(13)T	The **vrf** keyword and *vrf-name* argument were added.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(18)SXE	Support for this command was introduced on the Supervisor Engine 720.
12.2(27)SBC	This command was integrated into Cisco IOS Release 12.2(27)SBC.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use this command if you want the router to send SA messages for sources active in the PIM dense mode region to MSDP peers.

Specifying the interface-type and interface-number values allow the MSDP peers to forward source-active messages away from this border. The IP address of the interface is used as the originator ID, which is the rendezvous point field in the MSDP source-active message.

Note We recommend configuring the border router in the sparse mode domain to proxy-register sources in the dense mode domain, and have the sparse mode domain use standard MSDP procedures to advertise these sources.

Note If you use this command, you must constrain the sources advertised by using the **ip msdp redistribute** command. Configure the **ip msdp redistribute** command to apply to only local sources. Be aware that this configuration can result in (S, G) state remaining long after a source in the dense mode domain has stopped sending.

Note The **ip msdp originator-id**command also identifies an interface type and number to be used as the RP address. If both the **ip msdp border**and **ip msdp originator-id**commands are configured, the address derived from the **ip msdp originator-id**command determines the address of the RP.

Examples

In the following example, the local router is not an RP. It borders a PIM sparse mode region with a dense mode region. It uses the IP address of Ethernet interface 0 as the "RP" address in SA messages.

```
ip msdp border sa-address ethernet0
```

Related Commands

Command	Description
ip msdp originator-id	Allows an MSDP speaker that originates an SA message to use the IP address of its interface as the RP address in the SA message.
ip msdp redistribute	Configures which (S, G) entries from the multicast routing table are advertised in SA messages originated to MSDP peers.

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 4: 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.
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 nhrp cache non-authoritative

To turn off authoritative flags on NHRP cache entries, use the **ip nhrp cache non-authoritative** command in interface configuration mode. To turn authoritative flags on again, use the no form of this command.

ip nhrp cache non-authoritative
no ip nhrp cache non-authoritative

Syntax Description

This command has no arguments or keywords.

Command Default

Authoritative flags are turned on.

Command Modes

Interface configuration

Command History

Release	Modification
12.3(7)T	This command was introduced.

Usage Guidelines

By default the next hop server (NHS) replies to authoritative Next Hop Resolution Protocol (NHRP) resolution requests if it has a cache entry that is marked as authoritative. The **ip nhrp cache non-authoritative** comand turns off the “authoritative” flag on the cache entries. Thus, the request is forwarded to the next hop client (NHC), which responds to the resolution.

Configuring the **ip nhrp cache non-authoritative**command offloads the resolution replies from the hub to the spokes. It also helps the spokes complete NHRP mapping entries when a spoke-to-spoke tunnel is built, thus alleviating flap conditions in which the IP security (IPsec) tunnel is built but for which there are no corresponding NHRP mappings.

Examples

The following example shows that the authoritative flags have been turned off:

```
interface Tunnel0
 ip nhrp cache non-authoritative
```

ip nhrp nhs

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

Cisco IOS Release 12.2(33)SRA, 12.2SX, and Later Releases
ip nhrp nhs *nhs-address* [*net-address* [*netmask*]]
no ip nhrp nhs *nhs-address* [*net-address* [*netmask*]]

Cisco IOS Release 15.1(2)T and Later Releases
ip nhrp nhs {*nhs-address* [**nbma** {*nbma-addressFQDN-string*}] [**multicast**] [**priority** *value*] [**cluster** *value*] | **cluster** *value* **max-connections** *value* | **dynamic nbma** {*nbma-addressFQDN-string*} [**multicast**] [**priority** *value*] [**cluster** *value*] | **fallback** *seconds*}
no ip nhrp nhs {*nhs-address* [**nbma** {*nbma-addressFQDN-string*}] [**multicast**] [**priority** *value*] [**cluster** *value*] | **cluster** *value* **max-connections** *value* | **dynamic nbma** {*nbma-addressFQDN-string*} [**multicast**] [**priority** *value*] [**cluster** *value*] | **fallback** *seconds*}

Syntax Description

nhs-address	Address of the next-hop server being specified.
net-address	(Optional) IP address of a network served by the next-hop server.
netmask	(Optional) IP network mask to be associated with the IP address. The IP address is logically ANDed with the mask.
nbma	(Optional) Specifies the nonbroadcast multiple access (NBMA) address or FQDN.
nbma-address	NBMA address.
FQDN-string	Next hop server (NHS) fully qualified domain name (FQDN) string.
multicast	(Optional) Specifies to use NBMA mapping for broadcasts and multicasts.
priority *value*	(Optional) Assigns a priority to hubs to control the order in which spokes select hubs to establish tunnels. The range is from 0 to 255; 0 is the highest and 255 is the lowest priority.
cluster *value*	(Optional) Specifies NHS groups. The range is from 0 to 10; 0 is the highest and 10 is the lowest. The default value is 0.
max-connections *value*	Specifies the number of NHS elements from each NHS group that needs to be active. The range is from 0 to 255.
dynamic	Configures the spoke to learn the NHS protocol address dynamically.
fallback *seconds*	Specifies the duration, in seconds, for which the spoke must wait before falling back to an NHS of higher priority upon recovery.

Command Default

No next-hop servers are explicitly configured, so normal network layer routing decisions are used to forward NHRP traffic.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
10.3	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.
15.1(2)T	This command was modified. The *net-address* and *mask* arguments were removed and the **nbma**, *nbma-address*, *FQDN-string*, **multicast**, **priority** *value*, **cluster** *value*, **max-connections** *value*, **dynamic**, and **fallback** *seconds* keywords and arguments were added.
15.2(1)T	This command was modified. The NBMA address was modified to support IPv6 address.

Usage Guidelines

Use the **ip nhrp nhs** 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.

When the **ip nhrp nhs dynamic** command is configured on a DMVPN tunnel and the **shut** command is issued to the tunnel interface, the crypto socket does not receive shut message, thereby not bringing up a DMVPN session with the hub.

For any next hop server that is configured, you can specify multiple networks by repeating this command with the same *nhs-address*argument, but with different IP network addresses.

Examples

The following example shows how to register a hub to a spoke using NBMA and FQDN:

```
Router# configure terminal
Router(config)# interface tunnel 1
Router(config-if)# ip nhrp nhs 192.0.2.1 nbma examplehub.example1.com
```

The following example shows how to configure the desired **max-connections** value:

```
Router# configure terminal
Router(config)# interface tunnel 1
Router(config-if)# ip nhrp nhs cluster 5 max-connections 100
```

The following example shows how to configure the NHS fallback time:

```
Router# configure terminal
Router(config)# interface tunnel 1
Router(config-if)# ip nhrp nhs fallback 25
```

The following example shows how to configure NHS priority and group values:

```
Router# configure terminal
Router(config)# interface tunnel 1
Router(config-if)# ip nhrp nhs 192.0.2.1 priority 1 cluster 2
```

Related Commands

Command	Description
ip nhrp map	Statically configures the IP-to-NBMA address mapping of IP destinations connected to an NBMA network.
show ip nhrp	Displays NHRP mapping information.

ip port-map

To establish port-to-application mapping (PAM), use the **ip port-map** command in global configuration mode. To delete user-defined PAM entries, use the **no** form of this command.

ip port-map *appl-name* **port** [**tcp** | **udp**] [*port-num* | **from** *begin-port-num* **to** *end-port-num*] [**list** {*standard-acl-numberextended-acl-number ipv6-acl*}] [**description** *description-string*]
no ip port-map *appl-name* **port** [**tcp** | **udp**] [*port-num* | **from** *begin-port-num* **to** *end-port-num*] [**list** {*standard-acl-numberextended-acl-number ipv6-acl*}] [**description** *description-string*]

Syntax Description

appl-name	The application used to apply the port mapping. An application name can contain an underscore or a hyphen. An application can also be system or user-defined. However, a user-defined application must have the prefix *user-* in it; for example, *user-payroll*, *user-sales*, or *user-10*. Otherwise, the following error message appears: "Unable to add port-map entry. Names for user-defined applications must start with 'user-'."
port	Indicates that a port number maps to the application. You can specify up to five port numbers for each port.
tcp | **udp**	(Optional) Specifies the protocol for the application. For well-known applications (and those existing under PAM), you can omit these keywords, and the system configures the standard protocol for that application. However, for user-defined applications, you must specify either **tcp** or **udp**.
port-num	(Optional) The port number. The range is from 1 to 65535.
from *begin-port-num* **to** *end-port-num*	(Optional) Specifies a range of port numbers. You must use the **from** and **to** keywords together.
list	(Optional) Indicates that the port mapping information applies to a specific host or subnet by associating the port or subnet to an access control list (ACL) number used with PAM.
standard-acl-number	(Optional) The standard ACL number. The range is from 1 to 99.
extended-acl-number	(Optional) The extended ACL number. The range is from 1300 to 1999.
ipv6-acl	(Optional) Name of the IPv6 ACL.
description *description-string*	(Optional) Specifies a description of up to 40 characters in length.

Command Default

PAM does not get established by default.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.3(1)	This command was modified. Support for the Skinny Client Control Protocol (SCCP) was added.
12.3(14)T	This command was modified. Support was added for the following: • User-defined application names • User-specified descriptions • Port ranges • **tcp** and **udp** keywords • **from** *begin-port-num* **to** *end-port-num* keyword-argument pairs • **description** *description-string* keyword-argument pair
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.(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 3.6S Release	This command was modified. The *ipv6-acl* argument was added.

Usage Guidelines

The **ip port-map** command associates TCP or UDP port numbers with applications or services, establishing a table of default port mapping information at the firewall. The port mapping information is used to support network environments that run services using ports that are different from the registered or well-known ports associated with a service or application.

When you configure the **no** form of the command, include all the parameters needed to remove the entry matching that specific set of parameters. For example, when you configure the **no ip port-map** *appl-name* command, all entries for that application are removed.

The port mapping information in the PAM table can be one of the following three types:

- System-defined
- User-defined
- Host-specific

System-Defined Port Mapping

Initially, PAM creates a set of system-defined entries in the mapping table by using well-known or registered port mapping information set up during system startup. The Context-Based Access Control (CBAC) firewall requires the system-defined port mapping information to function.

You can delete or modify system-defined port mapping information. Use the **no** form of the command to delete a port mapping and the regular form of the command to remap the system-defined port mapping information to another application.

You can also add new port numbers to system-defined applications. However, for some system-defined applications like HTTP and Simple Mail Transfer Protocol (SMTP), in which the firewall inspects deeper into packets, the protocol (UDP or TCP) cannot be changed from that defined in the system. In these instances, error messages are displayed.

The table below lists some default system-defined services and applications in the PAM table. (Use the **show ip port-map** command to display the complete list.)

Table 5: System-Defined Port Mapping

Application Name	Well-Known or Registered Port Number	Protocol Description
cuseeme	7648	CU-SeeMe Protocol
exec	512	Remote Process Execution
ftp	21	File Transfer Protocol (control port)
h323	1720	H.323 Protocol (for example, MS NetMeeting, Intel Video Phone)
http	80	Hypertext Transfer Protocol
login	513	Remote login
msrpc	135	Microsoft Remote Procedure Call
netshow	1755	Microsoft NetShow
real-audio-video	7070	RealAudio and RealVideo
sccp	2000	Skinny Client Control Protocol
smtp	25	Simple Mail Transfer Protocol
sql-net	1521	SQL-NET
streamworks	1558	StreamWorks Protocol
sunrpc	111	SUN Remote Procedure Call
tftp	69	Trivial File Transfer Protocol
vdolive	7000	VDOLive Protocol

Note You can override system-defined entries for a specific host or a subnet using the **ip port-map** *appl-name* **port list** command.

User-Defined Port Mapping

Network applications that use nonstandard ports require user-defined entries in the mapping table. Use the **ip port-map** command to create default user-defined entries in the PAM table. These entries automatically appear as an option for the **ip inspect name** command to facilitate the creation of inspection rules.

You can specify up to five separate port numbers for each port map in a single entry. You can also specify a port range in a single entry. However, you cannot specify both single port numbers and port ranges in the same entry.

Note If you try to map an application to a system-defined port, a message appears warning you of a mapping conflict. Delete the system-defined entry before mapping it to another application. Deleted system-defined mappings appear in the running configuration in their **no ip port-map** form.

Use the **no** form of the **ip port-map** command to delete user-defined entries from the PAM table. To remove a single mapping, use the **no** form of the command with all its parameters.

To overwrite an existing user-defined port mapping, use the **ip port-map** command to associate another service or application with the specific port.

Multiple commands for the same application name are cumulative.

If you assign the same port number to a new application, the new entry replaces the existing entry. The entry no longer appears in the running configuration and you receive a message about the remapping.

You cannot specify a port number that is in a range assigned to another application. You cannot specify overlapping port ranges.

Host-Specific Port Mapping

User-defined entries in the mapping table can include host-specific mapping information, which establishes port mapping information for specific hosts or subnets. In some environments, it might be necessary to override the default port mapping information for a specific host or a subnet, including system-defined default port mapping information. Use the **ip port-map** *appl-name* **port list** command to specify an ACL for a host or a subnet that uses PAM.

Note If the host-specific port mapping information is the same as the existing system-defined or user-defined default entries, host-specific port changes have no effect.

Examples

The following examples show how to add and remove user-defined PAM configuration entries at the firewall.

The following example shows how to establish the nonstandard port 8000 as the user-defined default port for HTTP services:

```
Device(config)# ip port-map http port 8000
```

The following example shows how to configure PAM entries that establish a range of nonstandard ports for HTTP services:

```
Device(config)# ip port-map http port 8001
Device(config)# ip port-map http port 8002
```

```
Device(config)# ip port-map http port 8003
Device(config)# ip port-map http port 8004
```

The following example shows how to configure port 8000 for FTP services. ACL 10 identifies the server address (192.168.32.43), and port 8000 is mapped with FTP services:

```
Device(config)# access-list 10 permit 192.168.32.43
Device(config)# ip port-map ftp port 8000 list 10
```

The following example shows how to configure port 21, which is usually reserved for FTP services, to the RealAudio application for hosts in the ACL list 10. In this configuration, hosts in list 10 do not recognize FTP activity on port 21.

```
Device(config)# ip port-map realaudio port 21 list 10
```

The following example shows that the **ip port-map** command has failed and an error message is generated:

```
Device(config)# ip port-map netshow port 21

Command fail: the port 21 has already been defined for ftp by the system.
              No change can be made to the system defined port mappings.
```

The following example shows how the **no** form of this command deletes user-defined entries from the PAM table. The **no** command has no effect on the system-defined port mappings. This command deletes the host-specific port mapping of FTP.

```
Device(config)# no ip port-map ftp port 1022 list 10
```

Note All **no** forms of the **ip port-map** command appear before other entries in the running configuration.

The following example shows how to configure a specific host to use port 8000 for FTP services. ACL 10 identifies the server address (192.168.32.43), and port 8000 is mapped with FTP services.

```
Device(config)# access-list 10 permit 192.168.32.43
Device(config)# ip port-map ftp port 8000 list 10
```

The following example shows how to configure a specific subnet to run HTTP services on port 8080. ACL 50 identifies the subnet, and the PAM entry maps port 8080 with HTTP services.

```
Device(config)# access-list 50 permit 192.168.92.0
Device(config)# ip port-map http port 8080 list 50
```

The following example shows how to configure a specific host to run HTTP services on port 25, which is the system-defined port number for SMTP services. This requires a host-specific PAM entry that overrides the system-defined default port mapping for HTTP, which is port 80. ACL 15 identifies the host address (192.168.33.43), and port 25 is mapped with HTTP services.

```
Device(config)# access-list 15 permit 192.168.33.43
Device(config)# ip port-map http port 25 list 15
```

The following example shows how to configure the same port number for different services running on different hosts. Port 8000 is required for HTTP services by host 192.168.3.4, and also required for FTP services by host 192.168.5.6. ACL 10 and ACL 20 identify the specific hosts, and PAM maps the ports with the services for each ACL.

```
Device(config)# access-list 10 permit 192.168.3.4
Device(config)# access-list 20 permit 192.168.5.6
Device(config)# ip port-map http port 8000 list 10
Device(config)# ip port-map http port 8000 list 20
```

The following example shows how to configure five separate port numbers:

```
Device(config)# ip port-map user-my-app port tcp 8085 8087 8092 8093 8094
```

The following example shows how to configure multiple commands for the same application name and both ports map to the myapp application:

```
Device(config)# ip port-map user-myapp port tcp 3400
Device(config)# ip port-map user-myapp port tcp 3500
```

The following example shows how to configure the same port number for a new application. The new entry replaces the existing entry, meaning that port 5670 gets mapped to user-my-new-app and its mapping to myapp is removed. As a result, the first command no longer appears in the running configuration and you receive a message about the remapping.

```
Device(config)# ip port-map user-myapp port tcp 5670
Device(config)# ip port-map user-my-new-app port tcp 5670
```

In the following example, the second command assigns port 8085 to user-my-new-app because you cannot specify a port number that is in a range assigned to another application. As a result, the first command no longer appears in the running configuration, and you receive a message about the port being moved from one application to another.

```
Device(config)# ip port-map user-my-app port tcp 8085
Device(config)# ip port-map user-my-new-app port tcp from 8080 to 8090
```

Similarly, in the following example the second command assigns port range 8080 to 8085 to user-my-new-app and the first command no longer appears in the running configuration. You receive a message about the remapping.

```
Device(config)# ip port-map user-my-app port tcp from 8080 to 8085
Device(config)# ip port-map user-my-new-app port tcp from 8080 to 8090
```

Related Commands

Command	Description
show ip port-map	Displays PAM information.

ip radius source-interface

To force RADIUS to use the IP address of a specified interface for all outgoing RADIUS packets, use the **ip radius source-interface**command in global configuration mode. To prevent RADIUS from using the IP address of a specified interface for all outgoing RADIUS packets, use the no form of this command.

ip radius source-interface *subinterface-name* [**vrf** *vrf-name*]
no ip radius source-interface

Syntax Description

subinterface-name	Name of the interface that RADIUS uses for all of its outgoing packets.
vrf *vrf-name*	(Optional) Per virtual route forwarding (VRF) configuration.

Command Default

No default behavior or values.

Command Modes

Global configuration (config)

Command History

Release	Modification
11.3	This command was introduced.
12.2(1)DX	The vrf keyword and *vrf-name* argument were implemented on the Cisco 7200 series and Cisco 7401ASR.
12.2(2)DD	This command was integrated into Cisco IOS Release 12.2(2)DD.
12.2(4)B	This command was integrated into Cisco IOS Release 12.2(4)B.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
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.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.

Usage Guidelines

Use this command to set the IP address of a subinterface to be used as the source address for all outgoing RADIUS packets. The IP address is used as long as the subinterface is in the *up* state. The RADIUS server can use one IP address entry for every network access client instead of maintaining a list of IP addresses. Radius uses the IP address of the interface that it is associated to, regardless of whether the interface is in the *up* or *down* state.

The **ip radius source-interface**command is especially useful in cases where the router has many subinterfaces and you want to ensure that all RADIUS packets from a particular router have the same IP address.

The specified sub-interface should have a valid IP address and should be in the *up* state for a valid configuration. If the specified sub-interface does not have a valid IP address or is in the *down* state, RADIUS enforces the source-interface configuration. In case the interface has no IP address, RADIUS configures the best available local IP address. To avoid this, add a valid IP address to the sub-interface or bring the sub-interface to the *up* state.

Use the **vrf** *vrf-name* keyword and argument to configure this command per VRF, which allows multiple disjoined routing or forwarding tables, where the routes of one user have no correlation with the routes of another user.

Examples

The following example shows how to configure RADIUS to use the IP address of subinterface s2 for all outgoing RADIUS packets:

```
ip radius source-interface s2
```

The following example shows how to configure RADIUS to use the IP address of subinterface Ethernet0 for VRF definition:

```
ip radius source-interface Ethernet0 vrf vrf1
```

Related Commands

Command	Description
ip tacacs source-interface	Uses the IP address of a specified interface for all outgoing TACACS packets.
ip telnet source-interface	Allows a user to select an address of an interface as the source address for Telnet connections.
ip tftp source-interface	Allows a user to select the interface whose address will be used as the source address for TFTP connections.

ip reflexive-list timeout

To specify the length of time that reflexive access list entries will continue to exist when no packets in the session are detected, use the **ip reflexive-list timeout** command in global configuration mode. To reset the timeout period to the default timeout, use the **no** form of this command.

ip reflexive-list timeout seconds
no ip reflexive-list timeout

Syntax Description

seconds	Specifies the number of seconds to wait (when no session traffic is being detected) before temporary access list entries expire. Use a positive integer from 0 to 2,147,483. The default is 300 seconds.

Command Default

300 seconds

Command Modes

Global configuration

Command History

Release	Modification
11.3	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

This command is used with reflexive filtering, a form of session filtering.

This command specifies when a reflexive access list entry will be removed after a period of no traffic for the session (the timeout period).

With reflexive filtering, when an IP upper-layer session begins from within your network, a temporary entry is created within the reflexive access list, and a timer is set. Whenever a packet belonging to this session is forwarded (inbound or outbound) the timer is reset. When this timer counts down to zero without being reset, the temporary reflexive access list entry is removed.

The timer is set to the *timeout period*. Individual timeout periods can be defined for specific reflexive access lists, but for reflexive access lists that do not have individually defined timeout periods, the global timeout period is used. The global timeout value is 300 seconds by default; however, you can change the global timeout to a different value at any time using this command.

This command does not take effect for reflexive access list entries that were already created when the command is entered; this command only changes the timeout period for entries created after the command is entered.

Examples

The following example sets the global timeout period for reflexive access list entries to 120 seconds:

```
ip reflexive-list timeout 120
```

The following example returns the global timeout period to the default of 300 seconds:

```
no ip reflexive-list timeout
```

Related Commands

Command	Description
evaluate	Nests a reflexive access list within an access list.
ip access-list	Defines an IP access list by name.
permit (reflexive)	Creates a reflexive access list and enables its temporary entries to be automatically generated.

ip route (vasi)

To establish a static route on the VRF-Aware Service Infrastructure (VASI) interface, use the **ip route vrf**command in global configuration mode. To remove the static route connection, use the **no** form of this command.

ip route [**vrf** *vrf-name*] *destination-prefix detination-prefix-mask* {**vasileft** | **vasiright**} *number*
no ip route [**vrf** *vrf-name*] *destination-prefix detination-prefix-mask* {**vasileft** | **vasiright**} *number*

Syntax Description

vrf *vrf-name*	Specifies the Virtual Routing and Forwarding (VRF) instance for the static route.
destination-prefix	IP route prefix for the destination, in dotted decimal format.
destination-prefix -mask	Prefix mask for the destination, in dotted decimal format.
vasileft	Configures the vasileft interface.
vasiright	Configures the vasiright interface.
number	Identifier of the VASI interface. The range is from 1 to 256.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Release 2.6	This command was introduced.

Examples

The following example shows how to configure static route on a VASI interface:

```
router(config)# ip route vrf red 0.0.0.0 0.0.0.0 vasileft 100
```

Related Commands

Command	Description
i**nterface (vasi)**	Configures the VASI interface.

debug interface (vasi)	Displays debugging information of VASI interface descriptor block.
debug vasi	Displays debugging information of VASI.
show vasi pair	Displays the status of a VASI pair.

ip scp password

To specify a password for SSH File Transfer Protocol (SCP), use the **ip scp password** command in global configuration mode. To remove the scp password, use the **no** form of this command.

ip scp password { **0** | **7** | **LINE** }
no ip scp password

0	Specifies an UNENCRYPTED password.
7	Specifies an ENCRYPTED password.
LINE	Specifies a clear text password.

Command Default

No password is defined.

Command Modes

Global configuration mode (config)

Command History

Release	Modification
Cisco IOS XE 17.14.1a	This command was introduced.

Examples

This example shows how to set a password for SCP (need example):

```
Device> enable
Device# configure terminal
Device(config)# ip scp password
```

ip scp server enable

To enable the router to securely copy files from a remote workstation, use the **ip scp server enable** command in global configuration mode. To disable secure copy functionality (the default), use the **no** form of this command.

ip scp server enable
no ip scp server enable

Syntax Description

This command has no arguments or keywords.

Command Default

The secure copy function is disabled.

Command Modes

Global configuration

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)S	This command was integrated into Cisco IOS Release 12.0(21)S and support for the Cisco 7500 series and Cisco 12000 series routers was added.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(15)S.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(33)SRA.

Usage Guidelines

Use this command to enable secure copying of files from systems using the Secure Shell (SSH) application. This secure copy function is accomplished by an addition to the **copy** command in the Cisco IOS software, which takes care of using the secure copy protocol (scp) to copy to and from a router while logged in to the router itself. Because copying files is generally a restricted operation in the Cisco IOS software, a user attempting to copy such files needs to be at the correct enable level.

The Cisco IOS software must also allow files to be copied to or from itself from a remote workstation running the SSH application (which is supported by both the Microsoft Windows and UNIX operating systems). To get this information, the Cisco IOS software must have authentication and authorization configured in the authentication, authorization, and accounting (AAA) feature. SSH already relies on AAA authentication to authenticate the user username and password. Scp adds the requirement that AAA authorization be turned on so that the operating system can determine whether or not the user is at the correct privilege level.

Examples

The following example shows a typical configuration that allows the router to securely copy files from a remote workstation. Because scp relies on AAA authentication and authorization to function properly, AAA must be configured.

```
aaa new-model
aaa authentication login default tac-group tacacs+
aaa authorization exec default local
```

```
username user1 privilege 15 password 0 lab
ip scp server enable
```

The following example shows how to use scp to copy a system image from Flash memory to a server that supports SSH:

```
Router# copy flash:c4500-ik2s-mz.scp scp://user1@host1/
Address or name of remote host [host1]?
Destination username [user1]?
Destination filename [c4500-ik2s-mz.scp]?
Writing c4500-ik2s-mz.scp
Password:
!!!
```

ip scp username

To specify a username for Secure Copy Protocol (SCP), use the **ip scp username** command in global configuration mode. To remove the scp username, use the **no** form of this command.

ip scp username *username*
no ip scp username

username	(Optional) Enter username

Command Default

No username is defined.

Command Modes

Global configuration mode (config)

Command History

Release	Modification
Cisco IOS XE 17.14.1a	This command was introduced.

Examples

This example shows how to set a password for SCP:

```
Device> enable
Device# configure terminal
Device(config)# ip scp username user1
```

ip sdee

To set the Security Device Event Exchange (SDEE) attribute values, use the **ip sdee** command in global configuration mode. To change the current selection or return to the default, use the **no** form of this command.

ip sdee {**alerts** *alert-number* | **messages** *message-number* | **subscriptions** *subscription-number*}
no ip sdee {**alerts** | **messages** | **subscriptions**}

Syntax Description

alerts *alert-number*	Specifies the maximum number of alerts the router must store. The range is from 10 to 2000. The default value is 200. **Note** Storing more alerts uses more router memory.
messages *message-number*	Specifies the maximum number of messages the router must store. The range is from 10 to 500. The default value is 200. **Note** Storing more messages uses more router memory.
subscriptions *subscription -number*	Specifies the maximum number of subscriptions. The range is from 1 to 3. The default value is 1.

Command Default

The default subscription is 1. The default message is 200. The default alert is 200.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(8)T	This command was introduced.
15.0(1)M	This command was modified in a release earlier than Cisco IOS Release 15.0(1)M. The **alerts** *alert-number* and **messages** *message-number* keywords and arguments were added.

Usage Guidelines

The SDEE messages report on the progress of Cisco IOS Intrusion Prevention System (IPS) initialization and operation. After you have enabled SDEE to receive and process events from IPS, you can issue the **ip sdee subscriptions** command to modify the number of allowed open SDEE subscriptions.

Examples

The following example shows how to change the number of allowed open subscriptions to 2:

```
Router# configure terminal
Router(config)# ip ips notify sdee
Router(config)# ip sdee events 500
Router(config)# ip sdee subscriptions 2
```

The following example shows how to change the number of alerts that must be stored on the router to 10:

```
Router# configure terminal
Router(config)# ip ips notify sdee
Router(config)# ip sdee events 500
Router(config)# ip sdee alerts 10
```

The following example shows how to change the number of messages that must be stored on the router to 10:

```
Router# configure terminal
Router(config)# ip ips notify sdee
Router(config)# ip sdee events 500
Router(config)# ip sdee messages 10
```

Related Commands

Command	Description
ip ips notify	Specifies the method of event notification.

ip sdee events

To set the maximum number of Security Device Event Exchange (SDEE) events that can be stored in the event buffer, use the **ip sdee events** command in global configuration mode. To change the buffer size or return to the default buffer size, use the **no** form of this command.

ip sdee events *events*
no ip sdee events *events*

Syntax Description

events	Maximum number of events; maximum number of allowable events: 1000.

Command Default

200 events

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

When SDEE notification is enabled (via the **ip ips notify sdee** command), 200 hundred events can automatically be stored in the buffer. When SDEE notification is disabled, all stored events are lost. A new buffer is allocated when the notifications are reenabled.

When specifying the size of an events buffer, note the following functionality:

- It is circular. When the end of the buffer is reached, the buffer will start overwriting the earliest stored events. (If overwritten events have not yet been reported, you will receive a buffer overflow notice.)
- If a new, smaller buffer is requested, all events that are stored in the previous buffer will be lost.
- If a new, larger buffer is requested, all existing events will be saved.

Examples

The following example shows how to set the maximum buffer events size to 500:

```
configure terminal
 ip ips notify sdee
 ip sdee events 500
```

Related Commands

Command	Description
ip ips notify	Specifies the method of event notification.

ip security add

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

ip security add
no ip security add

Syntax Description

This command has no arguments or keywords.

Command Default

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

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

Examples

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

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

Related Commands

Command	Description
ip security dedicated	Sets the level of classification and authority on the interface.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.

Command	Description
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security aeso

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

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

Syntax Description

source	Extended Security Option (ESO) source. This can be an integer from 0 to 255.
compartment-bits	Number of compartment bits in hexadecimal.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

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

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

Examples

The following example defines the Extended Security Option source as 5 and sets the compartments bits to 5:

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

Related Commands

Command	Description
ip security eso-info	Configures system-wide defaults for extended IPSO information.
ip security eso-max	Specifies the maximum sensitivity level for an interface.
ip security eso-min	Configures the minimum sensitivity level for an interface.

Command	Description
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.

ip security dedicated

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

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

Syntax Description

level	Degree of sensitivity of information. The *level* keywords are listed in the first table below.
authority	Organization that defines the set of security levels that will be used in a network. The authority keywords are listed in the second table below.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

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

- level -- The degree of sensitivity of information. For example, data marked TOPSECRET is more sensitive than data marked SECRET. The level keywords and their corresponding bit patterns are shown in the table below.

Table 6: IPSO Level Keywords and Bit Patterns

Level Keyword	Bit Pattern
Reserved4	0000 0001
TopSecret	0011 1101
Secret	0101 1010
Confidential	1001 0110
Reserved3	0110 0110

Level Keyword	Bit Pattern
Reserved2	1100 1100
Unclassified	1010 1011
Reserved1	1111 0001

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

Table 7: IPSO Authority Keywords and Bit Patterns

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

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

Examples

The following example sets a confidential level with Genser authority:

```
ip security dedicated confidential Genser
```

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.

Command	Description
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security eso-info

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

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

Syntax Description

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

Command Default

Disabled

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

Examples

The following example sets system-wide defaults for source, compartment size, and the default bit value:

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

Related Commands

Command	Description
ip security eso-max	Specifies the maximum sensitivity level for an interface.
ip security eso-min	Configures the minimum sensitivity level for an interface.

ip security eso-max

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

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

Syntax Description

source	Extended Security Option (ESO) source. This is an integer from 1 to 255.
compartment-bits	Number of compartment bits in hexadecimal.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 command is used to specify the maximum sensitivity level for a particular interface. Before the per-interface compartment information for a particular Network-Level Extended Security Option (NLESO) source can be configured, the **ip security eso-info** global configuration command must be used to specify the default information.

On every incoming packet on the interface, these Extended Security Options should be present at the minimum level and should match the configured compartment bits. Every outgoing packet must have these ESOs.

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

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

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

Examples

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

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

Related Commands

Command	Description
ip security eso-info	Configures system-wide defaults for extended IPSO information.
ip security eso-min	Configures the minimum sensitivity level for an interface.

ip security eso-min

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

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

Syntax Description

source	Extended Security Option (ESO) source. This is an integer from 1 to 255.
compartment-bits	Number of compartment bits in hexadecimal.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 command is used to specify the minimum sensitivity level for a particular interface. Before the per-interface compartment information for a particular Network Level Extended Security Option (NLESO) source can be configured, the **ip security eso-info** global configuration command must be used to specify the default information.

On every incoming packet on this interface, these Extended Security Options should be present at the minimum level and should match the configured compartment bits. Every outgoing packet must have these ESOs.

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

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

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

Examples

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

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

Related Commands

Command	Description
ip security eso-info	Configures system-wide defaults for extended IPSO information.
ip security eso-max	Specifies the maximum sensitivity level for an interface.

ip security extended-allowed

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

ip security extended-allowed
no ip security extended-allowed

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

Packets containing extended security options are rejected.

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.
ip security multilevel	Sets the range of classifications and authorities on an interface.

Command	Description
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security first

To prioritize the presence of security options on a packet, use the **ip security first** command in interface configuration mode. To prevent packets that include security options from moving to the front of the options field, use the **no**form of this command.

ip security first
no ip security first

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.

Command	Description
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security ignore-authorities

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

ip security ignore-authorities
no ip security ignore-authorities

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.

Command	Description
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security ignore-cipso

To enable Cisco IOS software to ignore the Commercial IP Security Option (CIPSO) field of all incoming packets at the interface, use the **ip security ignore-cipso** command in interface configuration mode. To disable this function, use the **no** form of this command.

ip security ignore-cipso
no ip security ignore-cipso

Syntax Description

This command has no arguments or keywords.

Command Default

Cisco IOS software cannot ignore the CIPSO field.

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The **ip security ignore-cipso** command allows a router running Cisco IOS software to ignore the CIPSO field in the IP packet and forward the packet as if the field was not present.

Examples

The following example shows how to enable Cisco IOS software to ignore the CIPSO field for all incoming packets at the Ethernet interface:

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

The following sample output from the **show ip interface** command can be used to verify that the **ip security ignore-cipso** option has been enabled. If this option is enabled, the output will display the text “Commercial security options are ignored.”

```
Router# show ip interface ethernet 0
Ethernet0 is up, line protocol is up
Internet address is 172.16.0.0/28
Broadcast address is 255.255.255.255
Address determined by non-volatile memory
MTU is 1500 bytes
Helper address is not set
Directed broadcast forwarding is enabled
Secondary address 172.19.56.31/24
Outgoing access list is not set
Inbound access list is not set
Proxy ARP is enabled
Security level is default
Commercial security options are ignored
Split horizon is enabled
ICMP redirects are always sent
ICMP unreachables are always sent
ICMP mask replies are never sent
```

```
IP fast switching is enabled
IP fast switching on the same interface is disabled
IP multicast fast 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
Gateway Discovery is disabled
Policy routing is disabled
Network address translation is disabled
```

The following sample outputs from the **show ip traffic** command can be used to verify that the **ip security ignore-cipso** command has been enabled:

Sample Output Before the ip security ignore-cipso Command Was Introduced

```
Router# show ip traffic
IP statistics:
Rcvd: 153 total, 129 local destination
0 format errors, 0 checksum errors, 0 bad hop count
0 unknown protocol, 0 not a gateway
0 security failures, 34 bad options, 44 with options
Opts: 10 end, 0 nop, 0 basic security, 0 loose source route
0 timestamp, 0 extended security, 0 record route
0 stream ID, 0 strict source route, 0 alert, 0 other
Frags: 0 reassembled, 0 timeouts, 0 couldn't reassemble
0 fragmented, 0 couldn't fragment
Bcast: 108 received, 1 sent
Mcast: 0 received, 4 sent
Sent: 30 generated, 0 forwarded
2 encapsulation failed, 0 no route
Sample Output with the ip security ignore-cipso Command Enabled
Router# show ip traffic
IP statistics:
Rcvd: 153 total, 129 local destination
0 format errors, 0 checksum errors, 0 bad hop count
0 unknown protocol, 0 not a gateway
0 security failures, 34 bad options, 44 with options
Opts: 10 end, 0 nop, 0 basic security, 0 loose source route
0 timestamp, 0 extended security, 0 record route
0 stream ID, 0 strict source route, 0 alert, 44 cipso
0 other
Frags: 0 reas
sembled, 0 timeouts, 0 couldn't reassemble
0 fragmented, 0 couldn't fragment
Bcast: 108 received, 1 sent
Mcast: 0 received, 4 sent
Sent: 30 generated, 0 forwarded
2 encapsulation failed, 0 no route
```

Related Commands

Command	Description
show ip interfaces	Displays the usability status of interfaces configured for IP.
show ip traffic	Displays statistics about IP traffic.

ip security implicit-labelling

To force the Cisco IOS software to accept packets on the interface, even if they do not include a security option, use the **ip security implicit-labelling** command in interface configuration mode. To require security options, use the **no** form of this command.

Application Firewall Provisioning Syntax
ip security implicit-labelling [*level* *authority* [*authority...*]]
no ip security implicit-labelling [*level* *authority* [*authority...*]]

Syntax Description

level	(Optional) Degree of sensitivity of information. If your interface has multilevel security set, you must specify this argument. (See the *level* keywords listed in the first table in the **ip security dedicated** command section.)
authority	(Optional) Organization that defines the set of security levels that will be used in a network. If your interface has multilevel security set, you must specify this argument. You can specify more than one. (See the *authority* keywords listed in the second table in the **ip security dedicated** command section.)

Command Default

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

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.

Command	Description
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security multilevel

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

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

Syntax Description

level1	Degree of sensitivity of information. The classification level of incoming packets must be equal to or greater than this value for processing to occur. (See the *level* keywords found in the first table in the **ip security dedicated** command section.)
authority1	(Optional) Organization that defines the set of security levels that will be used in a network. The authority bits must be a superset of this value. (See the *authority* keywords listed in the second table in the **ip security dedicated**command section.)
to	Separates the range of classifications and authorities.
level2	Degree of sensitivity of information. The classification level of incoming packets must be equal to or less than this value for processing to occur. (See the *level* keywords found in the first table in the **ip security dedicated** command section.)
authority2	Organization that defines the set of security levels that will be used in a network. The authority bits must be a proper subset of this value. (See the *authority* keywords listed in the second table in the **ip security dedicated** command section.)

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

- The classification level must be greater than or equal to *level1* and less than or equal to *level2*.
- The authority bits must be a superset of *authority1* and a proper subset of *authority2*.That is, *authority1* specifies those authority bits that are required on a packet, and *authority2* specifies the required bits plus any optional authorities that also can be included. If the *authority1* field is the empty set, then a packet is required to specify any one or more of the authority bits in *authority*2.

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security reserved-allowed

To treat as valid any packets that have Reserved1 through Reserved4 security levels, use the **ip security reserved-allowed** command in interface configuration mode. To disallow packets that have security levels of Reserved3 and Reserved2, use the **no** form of this command.

ip security reserved-allowed
no ip security reserved-allowed

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.3	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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

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

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

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.

Command	Description
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security strip	Removes any basic security option on outgoing packets on an interface.

ip security strip

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

ip security strip
no ip security strip

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(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 removal procedure is performed after all security tests in the router have been passed. This command is not allowed for multilevel interfaces.

Examples

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

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

Related Commands

Command	Description
ip security add	Adds a basic security option to all outgoing packets.
ip security dedicated	Sets the level of classification and authority on the interface.
ip security extended-allowed	Accepts packets on an interface that has an Extended Security Option present.
ip security first	Prioritizes the presence of security options on a packet.
ip security ignore-authorities	Causes the Cisco IOS software to ignore the authorities field of all incoming packets.
ip security implicit-labelling	Forces the Cisco IOS software to accept packets on the interface, even if they do not include a security option.

Command	Description
ip security multilevel	Sets the range of classifications and authorities on an interface.
ip security reserved-allowed	Treats as valid any packets that have Reserved1 through Reserved4 security levels.

ip sftp password

To specify a password for SSH File Transfer Protocol (SFTP), use the **ip sftp password** command in global configuration mode. To remove the sftp password, use the **no** form of this command.

ip sftp password { **0** | **7** | **LINE** }
no ip sftp password

0	Specifies an UNENCRYPTED password.
7	Specifies an ENCRYPTED password.
LINE	Specifies a clear text password.

Command Default

No password is defined.

Command Modes

Global configuration mode (config)

Command History

Release	Modification
Cisco IOS XE 17.14.1a	This command was introduced.

Examples

This example shows how to set a password for SFTP:

```
Device> enable
Device# configure terminal
Device(config)# ip sftp password
```

ip sftp username

To specify a username for SSH File Transfer Protocol (SFTP), use the **ip sftp username** command in global configuration mode. To remove the sftp username, use the **no** form of this command.

ip sftp username *username*
no ip sftp username

username	(Optional) Enter username

Command Default

No username is defined.

Command Modes

Global configuration mode (config)

Command History

Release	Modification
Cisco IOS XE 17.14.1a	This command was introduced.

Examples

This example shows how to set a password for SFTP:

```
Device> enable
Device# configure terminal
Device(config)# ip sftp username user1
```