# Cisco IOS Security Command Reference: Commands D to L, Cisco IOS XE Release 3SE (Catalyst 3650 Switches)

**First Published:** January 11, 2013

**Last Modified:** January 11, 2013

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

# CONTENTS

# database archive through dns

# deny

To set conditions in a named IP access list or object group access control list (OGACL) that will deny packets, use the **deny** configuration command in the appropriate configuration mode. To remove a deny condition from an IP access list or OGACL, use the **no** form of this command.

**deny** *protocol* {*src-addr src-wildcard*| **object-group** *object-group-name*| **any**| **host** {*addr*| *name*}} {*dest-addr dest-wildcard*| **any**| **eq** *port*| **gt** *port*| **host** {*addr*| *name*}| **lt** *port*| **neq** *port*| **portgroup** *srcport-groupname*| **object-group** *dest-addr-groupname*| **range** *port*| [**dscp** *type*| **fragments**| **option** *option*| **precedence** *precedence*| **log**| **log-input**| **time-range** *time-range-name*| **tos** *tos*| **ttl** *ttl-value*]}

**no deny** *protocol* {*src-addr src-wildcard*| **object-group** *object-group-name*| **any**| **host** {*addr*| *name*}} {*dest-addr dest-wildcard*| **any**| **eq** *port*| **gt** *port*| **host** {*addr*| *name*}| **lt** *port*| **neq** *port*| **portgroup** *srcport-groupname*| **object-group** *dest-addr-groupname*| **range** *port*| [**dscp** *type*| **fragments**| **option** *option*| **precedence** *precedence*| **log**| **log-input**| **time-range** *time-range-name*| **tos** *tos*| **ttl** *ttl-value*]}

## Syntax Description

| | |
|---|---|
| *protocol* | Name or number of a protocol; valid values are **eigrp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including Internet Control Message Protocol (ICMP), TCP, and User Datagram Protocol (UDP), use the keyword **ip**. See the "Usage Guidelines" section for additional qualifiers. |
| *src-addr* | Number of the source network or host from which the packet is being sent in a 32-bit quantity in four-part, dotted-decimal format. |
| *src-wildcard* | Wildcard bits to be applied to source network in four-part, dotted-decimal format. Place ones in the bit positions you want to ignore. |
| object-group *object-group-name* | Specifies the source or destination name of the object group. |
| any | Specifies any source or any destination host as an abbreviation for the *source-addr*or *destination-addr value*and the *source-wildcard* or *destination-wildcard* value of 0.0.0.0 255.255.255.255. |
| host *addr* | Specifies the source or destination address of a single host. |
| host *name* | Specifies the source or destination name of a single host. |
| **tcp** | Specifies the TCP protocol. |

| | |
|---|---|
| **udp** | Specifies the UDP protocol. |
| object-group *source-addr-group-name* | Specifies the source address group name. |
| *destination-addr* | Number of the network or host to which the packet is being sent in a 32-bit quantity in four-part, dotted-decimal format. |
| *destination-wildcard* | Wildcard bits to be applied to the destination in a 32-bit quantity in four-part, dotted-decimal format. Place ones in the bit positions you want to ignore. |
| **eq** *port* | Matches only packets on a given port number; see the "Usage Guidelines" section for valid values. |
| **gt** *port* | Matches only the packets with a greater port number; see the "Usage Guidelines" section for valid values. |
| **lt** *port* | Matches only the packets with a lower port number; see the "Usage Guidelines" section for valid values. |
| **neq** *port* | Matches only the packets that are not on a given port number; see the "Usage Guidelines" section for valid values. |
| portgroup *srcport-group-name* | Specifies the source port object group name. |
| object-group *dest-addr-group-name* | Specifies the destination address group name. |
| portgroup *destport-group-name* | Specifies the destination port object group name. |
| **dscp** *type* | (Optional) Matches the packets with the given Differentiated Services Code Point (DSCP) value; see the "Usage Guidelines" section for valid values. |
| **fragments** | (Optional) Applies the access list entry to noninitial fragments of packets; the fragment is either permitted or denied accordingly. For more details about the **fragments** keyword, see the "Access List Processing of Fragments" and "deny, on page 2" sections in the "Usage Guidelines" section. |
| **option** *option* | (Optional) Matches the packets with the given IP options value number; see the "Usage Guidelines" section for valid values. |
| **precedence** *precedence* | (Optional) Specifies the precedence filtering level for packets; valid values are a number from 0 to 7 or by a name. See the "Usage Guidelines" section for a list of valid names. |

| | |
|---|---|
| **log** | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)<br>The message for a standard list includes the access list number, whether the packet was permitted or denied, the source address, and the number of packets.<br>The message for an extended list includes the access list number; whether the packet was permitted or denied; the protocol; whether the protocol was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers.<br>For both standard and extended lists, the message is generated for the first packet that matches, and then at 5-minute intervals, including the number of packets permitted or denied in the prior 5-minute interval.<br>The logging facility might drop some logging message packets if there are too many to be handled or if there is more than one logging message to be handled in 1 second. This behavior prevents the router from reloading because of too many logging packets. Therefore, the logging facility should not be used as a billing tool or an accurate source of the number of matches to an access list. |
| log-input | (Optional) Matches the log against this entry, including the input interface. |
| **time-range** *time-range-name* | (Optional) Specifies a time-range entry name. |
| **tos** *tos* | (Optional) Specifies the service filtering level for packets; valid values are a number from 0 to 15 or by a name as listed in the "Usage Guidelines" section of the **access-list**(IP extended) command. |
| **option** option | (Optional) Matches packets with the IP options value; see the "Usage Guidelines" section for the valid values. |
| **fragments** | (Optional) Applies the access list entry to noninitial fragments of packets; the fragment is either permitted or denied accordingly. For more details about the **fragments** keyword, see the deny, on page 2 and "deny, on page 2" sections in the "Usage Guidelines" section. |

| | |
|---|---|
| **ttl** *ttl-value* | (Optional) Matches packets with a given Time-to-live (ttl) value. |

**Command Default**

There is no specific condition under which a packet is denied passing the access list.

**Command Modes**

Standard access-list configuration (config-std-nacl) Extended access-list configuration (config-ext-nacl)

**Command History**

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

**Usage Guidelines**

Use this command following the **ip access-list** command to specify conditions under which a packet cannot pass the access list.

The **portgroup** keyword appears only when you configure an extended ACL.

The *address* or *object-group-name* value is created using the **object-group** command.

The **object-group** *object-group-name*keyword and argument allow you to create logical groups of users (or servers), which you can use to define access policy using ACLs. For example, with one ACL entry you can permit the object group named engineering to access all engineering servers. Otherwise, you would need one ACL entry for every person in the engineering group.

If the operator is positioned after the *source-addr* and *source-wildcard* values, it must match the source port.

If the operator is positioned after the *destination-addr* and *destination-wildcard* values, it must match the destination port.

If you are entering the port number of a TCP or UDP port, you can enter the decimal number or name of a TCP or UDP port. A port number is a number from 0 to 65535. TCP and UDP port names are listed in the "Usage Guidelines" section of the **access-list**(IP extended) command. TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP.

The valid values for the **dscp** *type* keyword and argument are as follows:

- 0 to 63--Differentiated services code point value.
- **af11** --Match packets with AF11 dscp (001010).
- **af12** --Match packets with AF12 dscp (001100).
- **af13** --Match packets with AF13 dscp (001110).
- **af21** --Match packets with AF21 dscp (010010).
- **af22** --Match packets with AF22 dscp (010100).
- **af23** --Matches the patches with the AF23 dscp (010110).
- **af31** --Matches the patches with the AF31 dscp (011010).
- **af32** --Matches the patches with the AF32 dscp (011100).

- **af33** --Matches the patches with the AF33 dscp (011110).
- **af41** --Matches the patches with the AF41 dscp (100010).
- **af42** --Matches the patches with the AF42 dscp (100100).
- **af43** --Matches the patches with the AF43 dscp (100110).
- **cs1** --Matches the patches with the CS1 (precedence 1) dscp (001000).
- **cs2** --Matches the patches with the CS2 (precedence 2) dscp (010000).
- **cs3** --Matches the patches with the CS3 (precedence 3) dscp (011000).
- **cs4** --Matches the patches with the CS4 (precedence 4) dscp (100000).
- **cs5** --Matches the patches with the CS5 (precedence 5) dscp (101000).
- **cs6** --Matches the patches with the CS6 (precedence 6) dscp (110000).
- **cs7** --Matches the patches with the CS7 (precedence 7) dscp (111000).
- **default** --Matches the patches with the default dscp (000000).
- **ef** --Matches the patches with the EF dscp (101110).

The valid values for the **eq** *port* keyword and argument are as follows:

- 0 to 65535--Port number.
- **bgp** --Border Gateway Protocol (179).
- **chargen** --Character generator (19).
- **cmd** --Remote commands (rcmd, 514).
- **daytime** --Daytime (13).
- **discard** --Discard (9).
- **domain** --Domain Name Service (53).
- **echo** --Echo (7).
- **exec** --Exec (rsh, 512).
- **finger** --Finger (79).
- **ftp** --File Transfer Protocol (21).
- **ftp-data** --FTP data connections (20).
- **gopher** --Gopher (70).
- **hostname** --NIC hostname server (101).
- **ident** --Ident Protocol (113).
- **irc** --Internet Relay Chat (194).
- **klogin** --Kerberos login (543).
- **kshell** --Kerberos shell (544).
- **login** --Login (rlogin, 513).

- **lpd** --Printer service (515).
- **nntp** --Network News Transport Protocol (119).
- **pim-auto-rp** --PIM Auto-RP (496).
- **pop2** --Post Office Protocol v2 (109).
- **pop3** --Post Office Protocol v3 (110).
- **smtp** --Simple Mail Transport Protocol (25).
- **sunrpc** --Sun Remote Procedure Call (111).
- **syslog** --Syslog (514).
- **tacacs** --TAC Access Control System (49).
- **talk** --Talk (517).
- **telnet** --Telnet (23).
- **time** --Time (37).
- **uucp** --Unix-to-Unix Copy Program (540).
- **whois** --Nicname (43).
- **www** --World Wide Web (HTTP, 80).

The valid values for the **gt** *port* keyword and argument are as follows:

- 0-65535--Port number.
- **biff** --Biff (mail notification, comsat, 512).
- **bootpc** --Bootstrap Protocol (BOOTP) client (68).
- **bootps** --Bootstrap Protocol (BOOTP) server (67).
- **discard** --Discard (9).
- **dnsix** --DNSIX security protocol auditing (195).
- **domain** --Domain Name Service (DNS, 53).
- **echo** --Echo (7).
- **isakmp** --Internet Security Association and Key Management Protocol (500).
- **mobile-ip** --Mobile IP registration (434).
- **nameserver** --IEN116 name service (obsolete, 42).
- **netbios-dgm** --NetBios datagram service (138).
- **netbios-ns** --NetBios name service (137).
- **netbios-ss** --NetBios session service (139).
- **non500-isakmp** --Internet Security Association and Key Management Protocol (4500).
- **ntp** --Network Time Protocol (123).
- **pim-auto-rp** --PIM Auto-RP (496).

- **rip** --Routing Information Protocol (router, in.routed, 520).
- **snmp** --Simple Network Management Protocol (161).
- **snmptrap** --SNMP Traps (162).
- sunrpc--Sun Remote Procedure Call (111).
- **syslog** --System Logger (514).
- **tacacs** --TAC Access Control System (49).
- **talk** --Talk (517).
- **tftp** --Trivial File Transfer Protocol (69).
- **time** --Time (37).
- **who** --Who service (rwho, 513).
- **xdmcp** --X Display Manager Control Protocol (177).

The valid values for the **lt** *port* keyword and argument are as follows:

- 0-65535--Port number.
- **biff** --Biff (mail notification, comsat, 512).
- **bootpc** --Bootstrap Protocol (BOOTP) client (68).
- **bootps** --Bootstrap Protocol (BOOTP) server (67).
- **discard** --Discard (9).
- **dnsix** --DNSIX security protocol auditing (195).
- **domain** --Domain Name Service (DNS, 53).
- **echo** --Echo (7).
- **isakmp** --Internet Security Association and Key Management Protocol (500).
- **mobile-ip** --Mobile IP registration (434).
- **nameserver** --IEN116 name service (obsolete, 42).
- **netbios-dgm** --NetBios datagram service (138).
- **netbios-ns** --NetBios name service (137).
- **netbios-ss** --NetBios session service (139).
- **non500-isakmp** --Internet Security Association and Key Management Protocol (4500).
- **ntp** --Network Time Protocol (123).
- **pim-auto-rp** --PIM Auto-RP (496).
- **rip** --Routing Information Protocol (router, in.routed, 520).
- **snmp** --Simple Network Management Protocol (161).
- **snmptrap** --SNMP Traps (162).
- **sunrpc** --Sun Remote Procedure Call (111).

- **syslog** --System Logger (514).
- **tacacs** --TAC Access Control System (49).
- **talk** --Talk (517).
- **tftp** --Trivial File Transfer Protocol (69).
- **time** --Time (37).
- **who** --Who service (rwho, 513).
- **xdmcp** --X Display Manager Control Protocol (177).

The valid values for the **neg** *port* keyword and argument are as follows:

- 0 to 65535--Port number.
- **biff** --Biff (mail notification, comsat, 512).
- **bootpc** --Bootstrap Protocol (BOOTP) client (68).
- **bootps** --Bootstrap Protocol (BOOTP) server (67).
- **discard** --Discard (9).
- **dnsix** --DNSIX security protocol auditing (195).
- **domain** --Domain Name Service (DNS, 53).
- **echo** --Echo (7).
- **isakmp** --Internet Security Association and Key Management Protocol (500).
- **mobile-ip** --Mobile IP registration (434).
- **nameserver** --IEN116 name service (obsolete, 42).
- **netbios-dgm** --NetBios datagram service (138).
- **netbios-ns** --NetBios name service (137).
- **netbios-ss** --NetBios session service (139).
- **non500-isakmp** --Internet Security Association and Key Management Protocol (4500).
- **ntp** --Network Time Protocol (123).
- **pim-auto-rp** --PIM Auto-RP (496).
- **rip** --Routing Information Protocol (router, in.routed, 520).
- **snmp** --Simple Network Management Protocol (161).
- **snmptrap** --SNMP Traps (162).
- **sunrpc** --Sun Remote Procedure Call (111).
- **syslog** --System Logger (514).
- **tacacs** --TAC Access Control System (49).
- **talk** --Talk (517).
- **tftp** --Trivial File Transfer Protocol (69).

- **time** --Time (37).
- **who** --Who service (rwho, 513).
- **xdmcp** --X Display Manager Control Protocol (177).

The valid values for the **option** *option* keyword and argument are as follows:

- 0 to 255--IP Options value.
- **add-ext** --Matches the packets with Address Extension Option (147).
- **any-options** --Matches the packets with ANY Option.
- **com-security** --Matches the packets with Commercial Security Option (134).
- **dps** --Matches the packets with Dynamic Packet State Option (151).
- **encode** --Matches the packets with Encode Option (15).
- **eool** --Matches the packets with End of Options (0).
- **ext-ip** --Matches the packets with the Extended IP Option (145).
- **ext-security** --Matches the packets with the Extended Security Option (133).
- **finn** --Matches the packets with the Experimental Flow Control Option (205).
    - **imitd**--Matches the packets with IMI Traffic Desriptor Option (144).
    - **lsr**--Matches the packets with Loose Source Route Option (131).
    - **match-all**--Matches the packets if all specified flags are present.
    - **match-any**--Matches the packets if any specified flag is present.
    - **mtup**--Matches the packets with MTU Probe Option (11).
    - **mtur**--Matches the packets with MTU Reply Option (12).
    - **no-op**--Matches the packets with No Operation Option (1).
    - **psh**--Match the packets on the PSH bit.
    - **nsapa**--Matches the packets with NSAP Addresses Option (150).
    - **reflect**--Creates reflexive access list entry.
    - **record-route**--Matches the packets with Record Route Option (7).
    - **rst**--Matches the packets on the RST bit.
    - **router-alert**--Matches the packets with Router Alert Option (148).
    - **sdb**--Matches the packets with Selective Directed Broadcast Option (149).
    - **security**--Matches the packets with Basic Security Option (130).
    - **ssr**--Matches the packets with Strict Source Routing Option (137).
    - **stream-id**--Matches the packets with Stream ID Option (136).
    - **syn**--Match the packets on the SYN bit.
- **timestamp** --Matches the packets with the Time Stamp Option (68).

- **traceroute** --Matches the packets with the Trace Route Option (82).
- **ump** --Matches the packets with the Upstream Multicast Packet Option (152).
- **visa** --Matches the packets with the Experimental Access Control Option (142).
- **zsu** --Matches the packets with the Experimental Measurement Option (10).

The valid values for the **tos** *value* keyword and argument are as follows:

- 0 to 15--Type of service value.
- **max-reliability** --Matches the packets with the maximum reliable ToS (2).
- **max-throughput** --Matches the packets with the maximum throughput ToS (4).
- **min-delay** --Matches the packets with the minimum delay ToS (8).
- **min-monetary-cost** --Matches packets with the minimum monetary cost ToS (1).
- **normal** --Matches the packets with the normal ToS (0).

### Access List or OGACL Processing of Fragments

The behavior of access-list entries regarding the use or lack of the **fragments** keyword are summarized in the table below:

*Table 1: Access list or OGACL Processing of Fragments*

| If the Access-List Entry Has... | Then... |
|---|---|
| ...no **fragments** keyword (the default behavior), and assuming all of the access-list entry information matches, | For an access-list entry containing only Layer 3 information:<br>• The entry is applied to nonfragmented packets, initial fragments, and noninitial fragments.<br>For an access list entry containing Layer 3 and Layer 4 information:<br>• The entry is applied to nonfragmented packets and initial fragments:<br>  • If the entry is a **permit**statement, the packet or fragment is permitted.<br>  • If the entry is a **deny**statement, the packet or fragment is denied.<br>• The entry is also applied to noninitial fragments in the following manner. Because noninitial fragments contain only Layer 3 information, only the Layer 3 portion of an access-list entry can be applied. If the Layer 3 portion of the access-list entry matches, and<br>  • If the entry is a **permit**statement, the noninitial fragment is permitted.<br>  • If the entry is a **deny** statement, the next access-list entry is processed.<br>**Note** The **deny** statements are handled differently for noninitial fragments versus nonfragmented or initial fragments. |
| ...the **fragments** keyword, and assuming all of the access-list entry information matches, | **Note** The access-list entry is applied only to noninitial fragments. The **fragments** keyword cannot be configured for an access-list entry that contains any Layer 4 information. |

Be aware that you should not simply add the **fragments** keyword to every access list entry because the first fragment of the IP packet is considered a nonfragment and is treated independently of the subsequent fragments. An initial fragment will not match an access list **permit** or **deny** entry that contains the **fragments**keyword, the packet is compared to the next access list entry, and so on, until it is either permitted or denied by an access list entry that does not contain the **fragments** keyword. Therefore, you may need two access list entries for every **deny** entry. The first **deny** entry of the pair will not include the **fragments** keyword, and applies to the initial fragment. The second **deny** entry of the pair will include the **fragments** keyword and applies to the

subsequent fragments. In the cases where there are multiple **deny** access-list entries for the same host but with different Layer 4 ports, a single **deny** access-list entry with the **fragments** keyword for that host is all that needs to be added. Thus all the fragments of a packet are handled in the same manner by the access list.

Packet fragments of IP datagrams are considered individual packets and each counts individually as a packet in access list accounting and access list violation counts.

**Note** The **fragments**keyword cannot solve all cases involving access lists and IP fragments.

**Fragments and Policy Routing**

Fragmentation and the fragment control feature affect policy routing if the policy routing is based on the **match ip address**s command and the access list had entries that match on Layer 4 through 7 information. It is possible that noninitial fragments pass the access list and are policy routed, even if the first fragment was not policy routed or the reverse.

By using the **fragments** keyword in access list entries as described earlier, a better match between the action taken for initial and noninitial fragments can be made and it is more likely policy routing will occur as intended.

The **portgroup** *srcport-groupname or* **portgroup** *destport-groupname*keywords and arguments allow you to create an object group based on a source or destination group.

## Examples

The following example creates an access list that denies all TCP packets:

```
Router> enable
Router# configure terminal
Router(config)# ip access-list extended my_ogacl_policy
Router(config-ext-nacl)# deny tcp any any
Router(config-ext-nacl)# exit
Router(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **ip access-group** | Applies an ACL or OGACL to an interface or a service policy map. |
| **ip access-list** | Defines an IP access list or OGACL by name or number. |
| **object-group network** | Defines network object groups for use in OGACLs. |
| **object-group service** | Defines service object groups for use in OGACLs. |
| **permit** | Sets conditions in a named IP access list or OGACL that will permit packets. |
| show ip access-list | Displays the contents of IP access lists or OGACLs. |

| Command | Description |
|---|---|
| show object-group | Displays information about object groups that are configured. |

# deny (IP)

To set conditions in a named IP access list that will deny packets, use the **deny** command in access list configuration mode. To remove a deny condition from an access list, use the **no** form of this command.

[ *sequence-number* ] **deny** *source* [ *source-wildcard* ]

[ *sequence-number* ] **deny** *protocol source source-wildcard destination destination-wildcard* [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**ttl** *operator value*] **[log]** [**time-range** *time-range-name*] **[fragments]**

**no** *sequence-number*

**no deny** *source* [ *source-wildcard* ]

**no deny** *protocol source source-wildcard destination destination-wildcard*

### Internet Control Message Protocol (ICMP)

[ *sequence-number* ] **deny icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* [ *icmp-code* ]| *icmp-message*] [**precedence** *precedence*] [**tos** *tos*] [**ttl** *operator value*] **[log]** [**time-range** *time-range-name*] **[fragments]**

### Internet Group Management Protocol (IGMP)

[ *sequence-number* ] **deny igmp** *source source-wildcard destination destination-wildcard* [ *igmp-type* ] [**precedence** *precedence*] [**tos** *tos*] [**ttl** *operator value*] **[log]** [**time-range** *time-range-name*] **[fragments]**

### Transmission Control Protocol (TCP)

**[sequence-number] deny tcp** *source source-wildcard* [*operator port* [ *port* ]] *destination destination-wildcard* [*operator* [ *port* ]] [**established** {**match-any**| **match-all**} {+-} *flag-name*| **precedence** *precedence*| **tos** *tos*| **ttl** *operator value*| **log**| **time-range** *time-range-name*| **fragments**]

### User Datagram Protocol (UDP)

[ *sequence-number* ] **deny udp** *source source-wildcard* [*operator port* [ *port* ]] *destination destination-wildcard* [*operator* [ *port* ]] [**precedence** *precedence*] [**tos** *tos*] [**ttl** *operator value*] **[log]** [**time-range** *time-range-name*] **[fragments]**

## Syntax Description

| | |
|---|---|
| *sequence-number* | (Optional) Sequence number assigned to the deny statement. The sequence number causes the system to insert the statement in that numbered position in the access list. |

| | |
|---|---|
| *source* | Number of the network or host from which the packet is being sent. There are three alternative ways to specify the source:<br>• Use a 32-bit quantity in four-part dotted-decimal format.<br>• Use the **any**keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use **host** *source*as an abbreviation for a *source*and *source-wildcard*of *source*0.0.0.0. |
| *source-wildcard* | Wildcard bits to be applied to the source . There are three alternative ways to specify the source wildcard:<br>• Use a 32-bit quantity in four-part dotted-decimal format. Place 1s in the bit positions that you want to ignore.<br>• Use the **any**keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use **host** *source*as an abbreviation for a *source*and *source-wildcard*of *source*0.0.0.0. |
| *protocol* | Name or number of an Internet protocol. The *protocol* argument can be one of the keywords **eigrp**, **gre**, **icmp**, **igmp**, **ip**, **ipinip**, **nos**, **ospf**, **tcp**, or **udp**, or an integer in the range from 0 to 255 representing an Internet protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the **ip**keyword.<br>**Note** When the **icmp**, **igmp**, **tcp,** and **udp** keywords are entered, they must be followed with the specific command syntax that is shown for the ICMP, IGMP, TCP, and UDP forms of the **deny** command. |
| **icmp** | Denies only ICMP packets. When you enter the **icmp** keyword, you must use the specific command syntax shown for the ICMP form of the **deny** command. |
| **igmp** | Denies only IGMP packets. When you enter the **igmp** keyword, you must use the specific command syntax shown for the IGMP form of the **deny** command. |
| **tcp** | Denies only TCP packets. When you enter the **tcp** keyword, you must use the specific command syntax shown for the TCP form of the **deny** command. |

| | |
|---|---|
| **udp** | Denies only UDP packets. When you enter the **udp** keyword, you must use the specific command syntax shown for the UDP form of the **deny** command. |
| *destination* | Number of the network or host to which the packet is being sent. There are three alternative ways to specify the destination:<br>• Use a 32-bit quantity in four-part dotted-decimal format.<br>• Use the **any**keyword as an abbreviation for the *destination*and *destination-wildcard*of 0.0.0.0 255.255.255.255.<br>• Use **host** *destination* as an abbreviation for a *destination*and *destination-wildcard* of *destination* 0.0.0.0. |
| *destination-wildcard* | Wildcard bits to be applied to the destination. There are three alternative ways to specify the destination wildcard:<br>• Use a 32-bit quantity in four-part dotted-decimal format. Place 1s in the bit positions that you want to ignore.<br>• Use the **any** keyword as an abbreviation for a *destination*and *destination-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use **host** *destination* as an abbreviation for a *destination*and *destination-wildcard* of *destination* 0.0.0.0. |
| **option** *option-name* | (Optional) Packets can be filtered by IP Options, as specified by a number from 0 to 255 or by the corresponding IP Option name, as listed in the table in the “Usage Guidelines” section. |
| **precedence** *precedence* | (Optional) Packets can be filtered by precedence level, as specified by a number from 0 to 7 or by a name. |
| **tos** *tos* | (Optional) Packets can be filtered by type of service (ToS) level, as specified by a number from 0 to 15, or by a name as listed in the “Usage Guidelines” section of the **access-list**(IP extended) command. |

| | |
|---|---|
| **ttl** *operator value* | (Optional) Compares the TTL value in the packet to the TTL value specified in this **deny** statement.<br>• The *operator* can be **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), or **range** (inclusive range).<br>• The *value* can range from 0 to 255.<br>• If the operator is **range**, specify two values separated by a space.<br>• For Release 12.0S, if the operator is **eq** or **neq**, only one TTL value can be specified.<br>• For all other releases, if the operator is **eq** or **neq**, as many as 10 TTL values can be specified, separated by a space. If the TTL in the packet matches just one of the possibly 10 values, the entry is considered to be matched. |
| **log** | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.) |
| **time-range** *time-range-name* | (Optional) Name of the time range that applies to this **deny** statement. The name of the time range and its restrictions are specified by the **time-range** and **absolute** or **periodic** commands, respectively. |
| **fragments** | (Optional) The access list entry applies to noninitial fragments of packets; the fragment is either permitted or denied accordingly. For more details about the **fragments** keyword, see the "deny (IP), on page 15" and "deny (IP), on page 15" sections in the "Usage Guidelines" section. |
| *icmp-type* | (Optional) ICMP packets can be filtered by ICMP message type. The type is a number from 0 to 255. |
| *icmp-code* | (Optional) ICMP packets that are filtered by ICMP message type can also be filtered by the ICMP message code. The code is a number from 0 to 255. |
| *icmp-message* | (Optional) ICMP packets can be filtered by an ICMP message type name or an ICMP message type and code name. The possible names are listed in the "Usage Guidelines" section of the **access-list**(IP extended) command. |

| | |
|---|---|
| *igmp-type* | (Optional) IGMP packets can be filtered by IGMP message type or message name. A message type is a number from 0 to 15. IGMP message names are listed in the "Usage Guidelines" section of the **access-list**(IP extended) command. |
| *operator* | (Optional) Compares source or destination ports. Operators include **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range).<br>If the operator is positioned after the source and source-wildcard arguments, it must match the source port. If the operator is positioned after the destination and destination-wildcard arguments, it must match the destination port.<br>The range operator requires two port numbers. Up to ten port numbers can be entered for the **eq** (equal) and **neq** (not equal) operators. All other operators require one port number. |
| *port* | (Optional) The decimal number or name of a TCP or UDP port. A port number is a number from 0 to 65535. TCP and UDP port names are listed in the "Usage Guidelines" section of the **access-list**(IP extended) command.<br>TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP. |
| **established** | (Optional) For the TCP protocol only: Indicates an established connection. A match occurs if the TCP datagram has the ACK or RST bit set. The nonmatching case is that of the initial TCP datagram to form a connection.<br>**Note** The **established** keyword can be used only with the old command-line interface (CLI) format. To use the new CLI format, you must use the **match-any** or **match-all** keywords followed by the **+** or **-** keywords and *flag-name* argument. |

| | |
|---|---|
| **match-any** \| **match-all** | (Optional) For the TCP protocol only: A match occurs if the TCP datagram has certain TCP flags set or not set. You use the **match-any**keyword to allow a match to occur if any of the specified TCP flags are present, or you can use the **match-all**keyword to allow a match to occur only if all of the specified TCP flags are present. You must follow the **match-any**and **match-all**keywords with the **+**or **-**keyword and the *flag-name*argument to match on one or more TCP flags. |
| **+** \| **-** *flag-name* | (Optional) For the TCP protocol only: The **+** keyword allows IP packets if their TCP headers contain the TCP flags that are specified by the *flag-name* argument. The **-** keyword filters out IP packets that do not contain the TCP flags specified by the *flag-name* argument. You must follow the **+** and **-** keywords with the *flag-name* argument. TCP flag names can be used only when filtering TCP. Flag names for the TCP flags are as follows: **urg**, **ack**, **psh**, **rst**, **syn**, and **fin**. |

**Command Default**

There are no specific conditions under which a packet is denied passing the named access list.

**Command Modes**

Access list configuration

**Command History**

| Release | Modification |
|---|---|
| 11.2 | This command was introduced. |
| 12.0(1)T | The **time-range** *time-range-name*keyword and argument were added. |
| 12.0(11) | The **fragments**keyword was added. |
| 12.2(13)T | The igrp keyword was removed because the IGRP protocol is no longer available in Cisco IOS software. |
| 12.2(14)S | The *sequence-number*argument was added. |
| 12.2(15)T | The *sequence-number*argument was added. |
| 12.3(4)T | The **option** *option-name* keyword and argument were added. The **match-any**, **match-all,** **+**,and **-**keywords and the *flag-name* argument were added. |
| 12.3(7)T | Command functionality was modified to allow up to ten port numbers to be added after the **eq** and **neq** operators so that an access list entry can be created with noncontiguous ports. |

| Release | Modification |
|---|---|
| 12.4(2)T | The **ttl** *operator value* keyword and arguments were added. |
| 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. |
| 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 following the **ip access-list** command to specify conditions under which a packet cannot pass the named access list.

The **time-range** keyword allows you to identify a time range by name. The **time-range**, **absolute**, and **periodic** commands specify when this **deny** statement is in effect.

### log Keyword

A log message includes the access list number, whether the packet was permitted or denied; the protocol, whether it was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers. The message is generated for the first packet that matches, and then at 5-minute intervals, including the number of packets permitted or denied in the prior 5-minute interval.

Use the **ip access-list log-update** command to generate logging messages when the number of matches reaches a configurable threshold (rather than waiting for a 5-minute-interval). See the **ip access-list log-update** command for more information.

The logging facility might drop some logging message packets if there are too many to be handled or if there is more than one logging message to be handled in 1 second. This behavior prevents the router from crashing because of too many logging packets. Therefore, the logging facility should not be used as a billing tool or an accurate source of the number of matches to an access list.

If you enable Cisco Express Forwarding (CEF) and then create an access list that uses the **log** keyword, the packets that match the access list are not CEF-switched. They are fast-switched. Logging disables CEF.

### Access List Filtering of IP Options

Access control lists can be used to filter packets with IP Options to prevent routers from being saturated with spurious packets containing IP Options. To see a complete table of all IP Options, including ones currently not in use, refer to the latest Internet Assigned Numbers Authority (IANA) information that is available from its URL: www.iana.org.

Cisco IOS software allows you to filter packets according to whether they contain one or more of the legitimate IP Options by entering either the IP Option value or the corresponding name for the *option-name* argument as shown in the table below.

***Table 2: IP Option Values and Names***

| IP Option Value or Name | Description |
|---|---|
| 0 to 255 | IP Options values. |

| IP Option Value or Name | Description |
|---|---|
| add-ext | Match packets with Address Extension Option (147). |
| any-options | Match packets with any IP Option. |
| com-security | Match packets with Commercial Security Option (134). |
| dps | Match packets with Dynamic Packet State Option (151). |
| encode | Match packets with Encode Option (15). |
| eool | Match packets with End of Options (0). |
| ext-ip | Match packets with Extended IP Options (145). |
| ext-security | Match packets with Extended Security Option (133). |
| finn | Match packets with Experimental Flow Control Option (205). |
| imitd | Match packets with IMI Traffic Descriptor Option (144). |
| lsr | Match packets with Loose Source Route Option (131). |
| mtup | Match packets with MTU Probe Option (11). |
| mtur | Match packets with MTU Reply Option (12). |
| no-op | Match packets with No Operation Option (1). |
| nsapa | Match packets with NSAP Addresses Option (150). |
| psh | Matches the packets on the PSH bit. |
| record-route | Match packets with Router Record Route Option (7). |
| reflect | Creates reflexive access list entry. |
| rst | Matches the packets on the RST bit. |
| router-alert | Match packets with Router Alert Option (148). |
| sdb | Match packets with Selective Directed Broadcast Option (149). |
| security | Match packets with Base Security Option (130). |

| IP Option Value or Name | Description |
|---|---|
| ssr | Match packets with Strict Source Routing Option (137). |
| stream-id | Match packets with Stream ID Option (136). |
| syn | Matches the packets on the SYN bit. |
| timestamp | Match packets with Time Stamp Option (68). |

### Filtering IP Packets Based on TCP Flags

The access list entries that make up an access list can be configured to detect and drop unauthorized TCP packets by allowing only the packets that have very specific groups of TCP flags set or not set. Users can select any desired combination of TCP flags with which to filter TCP packets. Users can configure access list entries in order to allow matching on a flag that is set and on a flag that is not set. Use the **+** and **-** keywords with a flag name to specify that a match is made based on whether a TCP header flag has been set. Use the **match-any** and **match-all** keywords to allow the packet if any or all, respectively, of the flags specified by the **+** or **-** keyword and *flag-name* argument have been set or not set.

### Access List Processing of Fragments

The behavior of access list entries regarding the use or lack of use of the **fragments** keyword can be summarized as follows:

| If the Access-List Entry Has... | Then... |
|---|---|
| ...no **fragments** keyword (the default behavior), and assuming all of the access-list entry information matches, | For an access list entry that contains only Layer 3 information:<br>• The entry is applied to nonfragmented packets, initial fragments, and noninitial fragments.<br><br>For an access list entry that contains Layer 3 and Layer 4 information:<br>• The entry is applied to nonfragmented packets and initial fragments.<br>  • If the entry is a **permit**statement, then the packet or fragment is permitted.<br>  • If the entry is a **deny**statement, then the packet or fragment is denied.<br>• The entry is also applied to noninitial fragments in the following manner. Because noninitial fragments contain only Layer 3 information, only the Layer 3 portion of an access list entry can be applied. If the Layer 3 portion of the access list entry matches, and<br>  • If the entry is a **permit**statement, then the noninitial fragment is permitted.<br>  • If the entry is a **deny** statement, then the next access list entry is processed.<br><br>**Note** The **deny** statements are handled differently for noninitial fragments versus nonfragmented or initial fragments. |
| ...the **fragments** keyword, and assuming all of the access-list entry information matches, | The access list entry is applied only to noninitial fragments. The **fragments** keyword cannot be configured for an access list entry that contains any Layer 4 information. |

Be aware that you should not add the **fragments** keyword to every access list entry because the first fragment of the IP packet is considered a nonfragment and is treated independently of the subsequent fragments. An initial fragment will not match an access list **permit** or **deny** entry that contains the **fragments**keyword. The packet is compared to the next access list entry, and so on, until it is either permitted or denied by an access list entry that does not contain the **fragments** keyword. Therefore, you may need two access list entries for every **deny** entry. The first **deny** entry of the pair will not include the **fragments** keyword and applies to the initial fragment. The second **deny** entry of the pair will include the **fragments** keyword and applies to the subsequent fragments. In the cases in which there are multiple **deny** access list entries for the same host but with different Layer 4 ports, a single **deny** access list entry with the **fragments** keyword for that host is all that needs to be added. Thus all the fragments of a packet are handled in the same manner by the access list.

Packet fragments of IP datagrams are considered individual packets, and each counts individually as a packet in access list accounting and access list violation counts.

**Note** The **fragments**keyword cannot solve all cases that involve access lists and IP fragments.

**Fragments and Policy Routing**

Fragmentation and the fragment control feature affect policy routing if the policy routing is based on the **match ip address** command and the access list has entries that match on Layer 4 through 7 information. It is possible that noninitial fragments pass the access list and are policy-routed, even if the first fragment is not policy-routed.

By using the **fragments** keyword in access list entries as described earlier, a better match between the action taken for initial and noninitial fragments can be made, and it is more likely that policy routing will occur as intended.

**Creating an Access List Entry with Noncontiguous Ports**

For Cisco IOS Release 12.3(7)T and later releases, you can specify noncontiguous ports on the same access control entry, which greatly reduces the number of access list entries required for the same source address, destination address, and protocol. If you maintain large numbers of access list entries, we recommend that you consolidate them when possible by using noncontiguous ports. You can specify up to ten port numbers following the **eq** and **neq** operators.

## Examples

The following example sets conditions for a standard access list named Internetfilter:

```
ip access-list standard Internetfilter
 deny 192.168.34.0  0.0.0.255
 permit 172.16.0.0  0.0.255.255
 permit 10.0.0.0  0.255.255.255
! (Note: all other access implicitly denied.)
```

The following example denies HTTP traffic on Monday through Friday from 8:00 a.m. to 6:00 p.m.:

```
time-range no-http
 periodic weekdays 8:00 to 18:00
!
ip access-list extended strict
 deny tcp any any eq http time-range no-http
!
interface ethernet 0
 ip access-group strict in
```

The following example adds an entry with the sequence number 25 to extended IP access list 150:

```
ip access-list extended 150
 25 deny ip host 172.16.3.3 host 192.168.5.34
```

The following example removes the entry with the sequence number 25 from the extended access list example shown above:

```
 no 25
```

The following example sets a deny condition for an extended access list named filter2. The access list entry specifies that a packet cannot pass the named access list if it contains the Strict Source Routing IP Option, which is represented by the IP option value ssr.

```
ip access-list extended filter2
 deny ip any any option ssr
```

The following example sets a deny condition for an extended access list named kmdfilter1. The access list entry specifies that a packet cannot pass the named access list if the RST and FIN TCP flags have been set for that packet:

```
ip access-list extended kmdfilter1
 deny tcp any any match-any +rst +fin
```

The following example shows several **deny** statements that can be consolidated into one access list entry with noncontiguous ports. The **show access-lists** command is entered to display a group of access list entries for the access list named abc.

```
Router# show access-lists abc
Extended IP access list abc
 10 deny tcp any eq telnet any eq 450
 20 deny tcp any eq telnet any eq 679
 30 deny tcp any eq ftp any eq 450
 40 deny tcp any eq ftp any eq 679
```

Because the entries are all for the same **deny** statement and simply show different ports, they can be consolidated into one new access list entry. The following example shows the removal of the redundant access list entries and the creation of a new access list entry that consolidates the previously displayed group of access list entries:

```
ip access-list extended abc
 no 10
 no 20
 no 30
 no 40
 deny tcp any eq telnet ftp any eq 450 679
```

The following examples shows the creation of the consolidated access list entry:

```
Router# show access-lists abc
Extended IP access list abc
 10 deny tcp any eq telnet ftp any eq 450 679
```

The following access list filters IP packets containing Type of Service (ToS) level 3 with TTL values 10 and 20. It also filters IP packets with a TTL greater than 154 and applies that rule to noninitial fragments. It permits IP packets with a precedence level of flash and a TTL not equal to 1, and sends log messages about such packets to the console. All other packets are denied.

```
ip access-list extended canton
 deny ip any any tos 3 ttl eq 10 20
 deny ip any any ttl gt 154 fragments
 permit ip any any precedence flash ttl neq 1 log
```

## Related Commands

| Command | Description |
|---|---|
| **absolute** | Specifies an absolute time when a time range is in effect. |
| **access-list (IP extended)** | Defines an extended IP access list. |
| **access-list (IP standard)** | Defines a standard IP access list. |
| **ip access-group** | Controls access to an interface. |
| **ip access-list** | Defines an IP access list by name. |

| Command | Description |
|---|---|
| **ip access-list log-update** | Sets the threshold number of packets that cause a logging message. |
| **ip access-list resequence** | Applies sequence numbers to the access list entries in an access list. |
| **ip options** | Drops or ignores IP Options packets that are sent to the router. |
| **logging console** | Sends system logging (syslog) messages to all available TTY lines and limits messages based on severity. |
| **match ip address** | Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, or performs policy routing on packets. |
| **periodic** | Specifies a recurring (weekly) time range for functions that support the time-range feature. |
| **permit (IP)** | Sets conditions under which a packet passes a named IP access list. |
| **remark** | Writes a helpful comment (remark) for an entry in a named IP access list. |
| **show access-lists** | Displays a group of access-list entries. |
| **show ip access-list** | Displays the contents of all current IP access lists. |
| **time-range** | Specifies when an access list or other feature is in effect. |

# deny (IPv6)

To set deny conditions for an IPv6 access list, use the **deny** command in IPv6 access list configuration mode. To remove the deny conditions, use the **no** form of this command.

**deny** *protocol* {*source-ipv6-prefix*/*prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [ *port-number* ]] {*destination-ipv6-prefix*/*prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [ *port-number* ]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] **[undetermined-transport]**

**no deny** *protocol* {*source-ipv6-prefix*/*prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [ *port-number* ]] {*destination-ipv6-prefix*/*prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [ *port-number* ]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] **[undetermined-transport]**

### Internet Control Message Protocol

**deny icmp** {*source-ipv6-prefix*/*prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [ *port-number* ]] {*destination-ipv6-prefix*/*prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [ *port-number* ]] [*icmp-type* [ *icmp-code* ]| *icmp-message*] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

### Transmission Control Protocol

**deny tcp** {*source-ipv6-prefix*/*prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [ *port-number* ]] {*destination-ipv6-prefix*/*prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [ *port-number* ]] **[ack]** [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] **[established] [fin]** [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**neq** {*port*| *protocol*}] **[psh]** [**range** {*port*| *protocol*}] **[routing]** [**routing-type** *routing-number*] **[rst]** [**sequence** *value*] **[syn]** [**time-range** *name*] **[urg]**

### User Datagram Protocol

**deny udp** {*source-ipv6-prefix*/*prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [ *port-number* ]] {*destination-ipv6-prefix*/*prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [ *port-number* ]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**neq** {*port*| *protocol*}] [**range** {*port*| *protocol*}] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

## Syntax Description

| | |
|---|---|
| *protocol* | Name or number of an Internet protocol. It can be one of the keywords **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **sctp**, **tcp**, **udp**, or **hbh**, or an integer in the range from 0 to 255 representing an IPv6 protocol number. |

| | |
|---|---|
| *source-ipv6-prefix*/*prefix-length* | The source IPv6 network or class of networks about which to set deny conditions.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **any** | An abbreviation for the IPv6 prefix ::/0. |
| **host** *source-ipv6-address* | The source IPv6 host address about which to set deny conditions.<br>This *source-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| *operator* [*port-number*] | (Optional) Specifies an operand that compares the source or destination ports of the specified protocol. Operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range).<br>If the operator is positioned after the *source-ipv6-prefix*/*prefix-length* argument, it must match the source port.<br>If the operator is positioned after the *destination-ipv6*/*prefix-length* argument, it must match the destination port.<br>The **range** operator requires two port numbers. All other operators require one port number.<br>The optional *port-number* argument is a decimal number or the name of a TCP or UDP port. A port number is a number from 0 to 65535. TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP. |
| *destination-ipv6-prefix*/*prefix-length* | The destination IPv6 network or class of networks about which to set deny conditions.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **host** *destination-ipv6-address* | The destination IPv6 host address about which to set deny conditions.<br>This *destination-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |

| | |
|---|---|
| **auth** | Allows matching traffic against the presence of the authentication header in combination with any protocol. |
| **dest-option-type** | (Optional) Matches IPv6 packets against the hop-by-hop option extension header within each IPv6 packet header. |
| *doh-number* | (Optional) Integer in the range from 0 to 255 representing an IPv6 destination option extension header. |
| *doh-type* | (Optional) Destination option header types. The possible destination option header type and its corresponding *doh-number* value are home-address—201. |
| **dscp** *value* | (Optional) Matches a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. |
| **flow-label** *value* | (Optional) Matches a flow label value against the flow label value in the Flow Label field of each IPv6 packet header. The acceptable range is from 0 to 1048575. |
| **fragments** | (Optional) Matches non-initial fragmented packets where the fragment extension header contains a non-zero fragment offset. The **fragments** keyword is an option only if the *operator* [*port-number*] arguments are not specified. |
| **hbh** | (Optional) Specifies a hop-by-hop options header. |
| **log** | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)<br>The message includes the access list name and sequence number, whether the packet was denied; the protocol, whether it was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers. The message is generated for the first packet that matches, and then at 5-minute intervals, including the number of packets denied in the prior 5-minute interval. |

| | |
|---|---|
| **log-input** | (Optional) Provides the same function as the **log** keyword, except that the logging message also includes the input interface. |
| **mobility** | (Optional) Extension header type. Allows matching of any IPv6 packet including a mobility header, regardless of the value of the mobility-header-type field within that header. |
| **mobility-type** | (Optional) Mobility header type. Either the *mh-number* or *mh-type* argument must be used with this keyword. |
| *mh-number* | (Optional) Integer in the range from 0 to 255 representing an IPv6 mobility header type. |
| *mh-type* | (Optional) Name of a mobility header type. Possible mobility header types and their corresponding *mh-number* value are as follows:<br>• 0—bind-refresh<br>• 1—hoti<br>• 2—coti<br>• 3—hot<br>• 4—cot<br>• 5—bind-update<br>• 6—bind-acknowledgment<br>• 7—bind-error |
| **routing** | (Optional) Matches source-routed packets against the routing extension header within each IPv6 packet header. |
| **routing-type** | (Optional) Allows routing headers with a value in the type field to be matched independently. The *routing-number* argument must be used with this keyword. |
| *routing-number* | Integer in the range from 0 to 255 representing an IPv6 routing header type. Possible routing header types and their corresponding *routing-number* value are as follows:<br>• 0—Standard IPv6 routing header<br>• 2—Mobile IPv6 routing header |

| | |
|---|---|
| **sequence** *value* | (Optional) Specifies the sequence number for the access list statement. The acceptable range is from 1 to 4294967295. |
| **time-range** *name* | (Optional) Specifies the time range that applies to the deny statement. The name of the time range and its restrictions are specified by the **time-range** and **absolute** or **periodic** commands, respectively. |
| **undetermined-transport** | (Optional) Matches packets from a source for which the Layer 4 protocol cannot be determined. The **undetermined-transport** keyword is an option only if the *operator* [*port-number*] arguments are not specified. |
| *icmp-type* | (Optional) Specifies an ICMP message type for filtering ICMP packets. ICMP packets can be filtered by ICMP message type. The ICMP message type can be a number from 0 to 255, some of which include the following predefined strings and their corresponding numeric values:<br>• 144—dhaad-request<br>• 145—dhaad-reply<br>• 146—mpd-solicitation<br>• 147—mpd-advertisement |
| *icmp-code* | (Optional) Specifies an ICMP message code for filtering ICMP packets. ICMP packets that are filtered by ICMP message type can also be filtered by the ICMP message code. The code is a number from 0 to 255. |
| *icmp-message* | (Optional) Specifies an ICMP message name for filtering ICMP packets. ICMP packets can be filtered by an ICMP message name or ICMP message type and code. The possible names are listed in the "Usage Guidelines" section. |
| **ack** | (Optional) For the TCP protocol only: acknowledgment (ACK) bit set. |
| **established** | (Optional) For the TCP protocol only: Indicates an established connection. A match occurs if the TCP datagram has the ACK or RST bits set. The nonmatching case is that of the initial TCP datagram to form a connection. |

| | |
|---|---|
| **fin** | (Optional) For the TCP protocol only: Fin bit set; no more data from sender. |
| **neq** {*port* \| *protocol*} | (Optional) Matches only packets that are not on a given port number. |
| **psh** | (Optional) For the TCP protocol only: Push function bit set. |
| **range** {*port* \| *protocol*} | (Optional) Matches only packets in the range of port numbers. |
| **rst** | (Optional) For the TCP protocol only: Reset bit set. |
| **syn** | (Optional) For the TCP protocol only: Synchronize bit set. |
| **urg** | (Optional) For the TCP protocol only: Urgent pointer bit set. |

## Command Default

No IPv6 access list is defined.

## Command Modes

IPv6 access list configuration (config-ipv6-acl)#

## Command History

| Release | Modification |
|---|---|
| 12.0(23)S | This command was introduced. |
| 12.2(13)T | This command was integrated into Cisco IOS Release 12.2(13)T. |
| 12.2(14)S | This command was integrated into Cisco IOS Release 12.2(14)S. |
| 12.4(2)T | The *icmp-type* argument was enhanced. The **dest-option-type**, **mobility**, **mobility-type**, and **routing-type** keywords were added. The *doh-number*, *doh-type*, *mh-number*, *mh-type*, and *routing-number* arguments were added. |
| 12.2(28)SB | This command was integrated into Cisco IOS Release 12.2(28)SB. |
| 12.2(25)SG | This command was integrated into Cisco IOS Release 12.2(25)SG. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| Cisco IOS XE Release 2.1 | This command was introduced on Cisco ASR 1000 Aggregation Series Routers. |

| Release | Modification |
|---|---|
| 12.4(20)T | The **auth** keyword was added. |
| 12.2(33)SRE | This command was integrated into Cisco IOS Release 12.2(33)SRE. |
| 15.2(3)T | This command was modified. Support was added for the **hbh** keyword. |
| 15.1(1)SY | This command was integrated into Cisco IOS Release 15.1(1)SY. |
| Cisco IOS XE Release 3.2SE | This command was integrated into Cisco IOS XE Release 3.2SE. |

## Usage Guidelines

The **deny** (IPv6) command is similar to the **deny** (IP) command, except that it is IPv6-specific.

Use the **deny** (IPv6) command following the **ipv6 access-list** command to define the conditions under which a packet passes the access list or to define the access list as a reflexive access list.

Specifying IPv6 for the *protocol* argument matches against the IPv6 header of the packet.

By 1default, the first statement in an access list is number 10, and the subsequent statements are numbered in increments of 10.

You can add **permit**, **deny**, **remark**, or **evaluate** statements to an existing access list without retyping the entire list. To add a new statement anywhere other than at the end of the list, create a new statement with an appropriate entry number that falls between two existing entry numbers to indicate where it belongs.

In Cisco IOS Release 12.2(2)T or later releases, 12.0(21)ST, and 12.0(22)S, IPv6 access control lists (ACLs) are defined and their deny and permit conditions are set by using the **ipv6 access-list** command with the **deny** and **permit** keywords in global configuration mode. In Cisco IOS Release 12.0(23)S or later releases, IPv6 ACLs are defined by using the **ipv6 access-list** command in global configuration mode and their permit and deny conditions are set by using the **deny** and **permit** commands in IPv6 access list configuration mode. Refer to the **ipv6 access-list** command for more information on defining IPv6 ACLs.

**Note** In Cisco IOS Release 12.0(23)S or later releases, every IPv6 ACL has implicit **permit icmp any any nd-na**, **permit icmp any any nd-ns**, and **deny ipv6 any any** statements as its last match conditions. (The former two match conditions allow for ICMPv6 neighbor discovery.) An IPv6 ACL must contain at least one entry for the implicit **deny ipv6 any any** statement to take effect. The IPv6 neighbor discovery process makes use of the IPv6 network layer service; therefore, by default, IPv6 ACLs implicitly allow IPv6 neighbor discovery packets to be sent and received on an interface. In IPv4, the Address Resolution Protocol (ARP), which is equivalent to the IPv6 neighbor discovery process, makes use of a separate data link layer protocol; therefore, by default, IPv4 ACLs implicitly allow ARP packets to be sent and received on an interface.

Both the *source-ipv6-prefix*/*prefix-length* and *destination-ipv6-prefix*/*prefix-length* arguments are used for traffic filtering (the source prefix filters traffic based upon the traffic source; the destination prefix filters traffic based upon the traffic destination).

**Note** IPv6 prefix lists, not access lists, should be used for filtering routing protocol prefixes.

The **fragments** keyword is an option only if the *operator* [*port-number*] arguments are not specified.

The **undetermined-transport** keyword is an option only if the *operator* [*port-number*] arguments are not specified.

The following is a list of ICMP message names:

- beyond-scope
- destination-unreachable
- echo-reply
- echo-request
- header
- hop-limit
- mld-query
- mld-reduction
- mld-report
- nd-na
- nd-ns
- next-header
- no-admin
- no-route
- packet-too-big
- parameter-option
- parameter-problem
- port-unreachable
- reassembly-timeout
- renum-command
- renum-result
- renum-seq-number
- router-advertisement
- router-renumbering
- router-solicitation
- time-exceeded
- unreachable

## Examples

The following example configures the IPv6 access list named toCISCO and applies the access list to outbound traffic on Ethernet interface 0. Specifically, the first deny entry in the list keeps all packets that have a destination

TCP port number greater than 5000 from exiting out of Ethernet interface 0. The second deny entry in the list keeps all packets that have a source UDP port number less than 5000 from exiting out of Ethernet interface 0. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets to exit out of Ethernet interface 0. The second permit entry in the list permits all other traffic to exit out of Ethernet interface 0. The second permit entry is necessary because an implicit deny all condition is at the end of each IPv6 access list.

```
ipv6 access-list toCISCO
 deny tcp any any gt 5000
 deny ::/0 lt 5000 ::/0 log
 permit icmp any any
 permit any any
interface ethernet 0
 ipv6 traffic-filter toCISCO out
```

The following example shows how to allow TCP or UDP parsing although an IPsec AH is present:

```
IPv6 access list example1
    deny tcp host 2001::1 any log sequence 5
    permit tcp any any auth sequence 10
    permit udp any any auth sequence 20
```

## Related Commands

| Command | Description |
|---|---|
| **ipv6 access-list** | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| **ipv6 traffic-filter** | Filters incoming or outgoing IPv6 traffic on an interface. |
| **permit (IPv6)** | Sets permit conditions for an IPv6 access list. |
| **show ipv6 access-list** | Displays the contents of all current IPv6 access lists. |

# dialer aaa

To allow a dialer to access the authentication, authorization, and accounting (AAA) server for dialing information, use the dialer aaa command in interface configuration mode. To disable this function, use the no form of this command.

**dialer aaa** [**password** *string*| **suffix** *string*]

**no dialer aaa** [**password** *string*| **suffix** *string*]

## Syntax Description

| | |
|---|---|
| **password** *string* | (Optional) Defines a nondefault password for authentication. The password string can be a maximum of 128 characters. |
| **suffix** *string* | (Optional) Defines a suffix for authentication. The suffix string can be a maximum of 64 characters. |

## Command Default

This feature is not enabled by default.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.0(3)T | This command was introduced. |
| 12.1(5)T | The password and **suffix** keywords were 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

This command is required for large scale dial-out and Layer 2 Tunneling Protocol (L2TP) dial-out functionality. With this command, you can specify a suffix, a password, or both. If you do not specify a password, the default password will be “cisco.”

**Note** Only IP addresses can be specified as usernames for the **dialer aaa suffix** command.

## Examples

This example shows a user sending out packets from interface Dialer1 with a destination IP address of 10.1.1.1. The username in the access-request message is “10.1.1.1@ciscoDoD” and the password is “cisco.”

```
interface dialer1
 dialer aaa
 dialer aaa suffix @ciscoDoD password cisco
```

## Related Commands

| Command | Description |
|---|---|
| **accept dialout** | Accepts requests to tunnel L2TP dial-out calls and creates an accept-dialout VPDN subgroup. |
| **dialer congestion-threshold** | Specifies congestion threshold in connected links. |
| **dialer vpdn** | Enables a Dialer Profile or DDR dialer to use L2TP dial-out. |

# dnsix-dmdp retries through dynamic

# domain (AAA)

To configure username domain options for the RADIUS application, use the **domain** command in dynamic authorization local server configuration mode. To disable the username domain options configured, use the **no** form of this command.

**domain** {**delimiter** *character*| **stripping [right-to-left]**}

**no domain** {**delimiter** *character*| **stripping [right-to-left]**}

## Syntax Description

| | |
|---|---|
| **delimiter** *character* | Specifies the domain delimiter. One of the following options can be specified: **@**, /, **$**, **%**, \, **#** or **-** |
| stripping | Compares the incoming username with the names oriented to the left of the **@** domain delimiter. |
| **right-to-left** | Terminates the string at the first delimiter going from right to left. |

## Command Default

No username domain options are configured.

## Command Modes

Dynamic authorization local server configuration (config-locsvr-da-radius)

## Command History

| Release | Modification |
|---|---|
| 12.2(31)SB14 | This command was introduced. |
| 12.2(33)SRC5 | This command was integrated into Cisco IOS Release 12.2(33)SRC5. |
| Cisco IOS XE Release 2.3 | This command was modified. This command was implemented on ASR 1000 series routers. |
| 15.1(2)T | This command was integrated into Cisco IOS Release 15.1(2)T. This command was also modified. The **right-to-left** keyword was added. |

## Usage Guidelines

If domain stripping is not configured, the full username provided in the authentication, authorization, and accounting (AAA) packet of disconnect (POD) messages is compared with the online subscribers. Configuring domain stripping allows you to send disconnect messages with only the username present before the **@** domain delimiter. The network access server (NAS) compares and matches this username with any online subscriber with a potential domain.

For instance, when domain stripping is configured and you send a POD message with the username “test,” a comparison between the POD message and online subscribers takes place, and subscribers with the username “test@cisco.com” or “test” match the specified username “test.”

## Examples

The following configuration example is used to match a username from right to left. If the username is user1@cisco.com@test.com, then the username to be matched by the POD message is user1@cisco.com.

```
Router# configure terminal
Router(config)# aaa server radius dynamic-author
Router(config-locsvr-da-radius)# domain stripping right-to-left
Router(config-locsvr-da-radius)# domain delimiter @
Router(config-locsvr-da-radius)# end
```

The following configuration example is used to match a username from left to right. If the username is user1@cisco.com@test.com, then the username to be matched by the POD message is user1.

```
Router# configure terminal
Router(config)# aaa server radius dynamic-author
Router(config-locsvr-da-radius)# domain stripping
Router(config-locsvr-da-radius)# domain delimiter @
Router(config-locsvr-da-radius)# end
```

## Related Commands

| Command | Description |
|---|---|
| **aaa server radius dynamic-author** | Configures a device as a AAA server to facilitate interaction with an external policy server. |

# dot1x control-direction

**Note** Effective with Cisco IOS Release 12.2(33)SXI, the **dot1x control-direction** command is replaced by the **authentication control-direction** command. See the **authentication control-direction** command for more information.

To change an IEEE 802.1X controlled port to unidirectional or bidirectional, use the **dot1x control-direction** command in interface configuration mode. To return to the default setting, use the **no** form of this command.

**dot1x control-direction** {**both**| **in**}

**no dot1x control-direction**

## Syntax Description

| | |
|---|---|
| **both** | Enables bidirectional control on the port. |
| **in** | Enables unidirectional control on the port. |

## Command Default

The port is set to bidirectional mode.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(25)SEC | This command was introduced. |
| 12.4(6)T | This command was integrated into Cisco IOS Release 12.4(6)T. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Switchs (ISRs) only. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.2(33)SXI | This command was replaced by the **authentication control-direction** command. |

## Usage Guidelines

The IEEE 802.1x standard defines a client-server-based access control and authentication protocol that restricts unauthorized devices from connecting to a LAN through publicly accessible ports. 802.1x controls network access by creating two distinct virtual access points at each port. One access point is an uncontrolled port; the other is a controlled port. All traffic through the single port is available to both access points. 802.1x authenticates each user device that is connected to a switch port and assigns the port to a VLAN before making

available any services that are offered by the switch or the LAN. Until the device is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL) traffic through the port to which the device is connected. After authentication is successful, normal traffic can pass through the port.

**Unidirectional State**

When you configure a port as unidirectional with the **dot1x control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state.

When Unidirectional Controlled Port is enabled, the connected host is in the sleeping mode or power-down state. The host does not exchange traffic with other devices in the network. The host connected to the unidirectional port cannot send traffic to the network, the host can only receive traffic from other devices in the network.

**Bidirectional State**

When you configure a port as bidirectional with the **dot1x control-direction both** interface configuration command, the port is access-controlled in both directions. In this state, the switch port receives or sends only EAPOL packets; all other packets are dropped.

Using the **both** keyword or using the **no** form of this command changes the port to its bidirectional default setting.

**Catalyst 6500 Series Switch**

Setting the port as bidirectional enables 802.1X authentication with wake-on-LAN (WoL).

**Cisco IOS Release 12.4(4)XC**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time; that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

## Examples

The following example shows how to enable unidirectional control:

```
Switch(config-if)# dot1x control-direction in
```

The following examples show how to enable bidirectional control:

```
Switch(config-if)# dot1x control-direction both
```

or

```
Switch(config-if)# no dot1x control-direction
```

You can verify your settings by entering the show dot1x all privileged EXEC command. The show dot1x all command output is the same for all devices except for the port names and the state of the port. If a host is attached to the port but is not yet authenticated, a display similar to the following appears:

```
Supplicant MAC 0002.b39a.9275
AuthSM State = CONNECTING
BendSM State = IDLE
PortStatus = UNAUTHORIZED
```

If you enter the dot1x control-direction in command to enable unidirectional control, the following appears in the show dot1x all command output:

```
ControlDirection = In
```

If you enter the dot1x control-direction in command and the port cannot support this mode because of a configuration conflict, the following appears in the show dot1x all command output:

```
ControlDirection = In (Disabled due to port settings):
```

The following example shows how to reset the global 802.1X parameters:

```
Switch(config)# dot1x default
```

## Examples

The following example shows how to enable 802.1X authentication with WoL and set the port as bidirectional:

```
Switch(config)# interface gigabitethernet 5/1
Switch(config-if)# dot1x control-direction both
```

## Examples

The following example shows Layer 3 802.1X support on a switched virtual interface (using a Cisco 870 ISR):

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
 dot1x reauthentication
 dot1x control-direction in
```

## Related Commands

| Command | Description |
|---|---|
| **show dot1x** | Displays details for an identity profile. |

# dot1x credentials

To specify which 802.1X credential profile to use when configuring a supplicant (client) or to apply a credentials structure to an interface and to enter dot1x credentials configuration mode, use the **dot1x credentials** command in global configuration or interface configuration mode. To remove the credential profile, use the **no** form of this command.

**dot1x credentials** *name*

**no dot1x credentials**

## Syntax Description

| | |
|---|---|
| *name* | Name of the credentials profile. |

## Command Default

A credentials profile is not specified.

## Command Modes

Global configuration Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.4(6)T | This command was introduced. |

## Usage Guidelines

An 802.1X credential structure is necessary when configuring a supplicant. This credentials structure may contain a username, password, and description.

## Examples

The following example shows which credentials profile should be used when configuring a supplicant:

```
dot1x credentials basic-user
 username router
 password secret
 description This credentials profile should be used for most configured ports
```

The credentials structure can be applied to an interface, along with the **dot1x pae supplicant** command and keyword, to enable supplicant functionality on that interface.

```
interface fastethernet 0/1
 dot1x credentials basic-user
 dot1x pae supplicant
```

## Related Commands

| Command | Description |
|---|---|
| **anonymous-id (dot1x credential)** | Specifies the anonymous identity that is associated with a credentials profile. |

| Command | Description |
|---|---|
| **description (dot1x credential)** | Specifies the description for an 802.1X credentials profile. |
| **password (dot1x credential)** | Specifies the password for an 802.1X credentials profile. |
| username (dot1x credential) | Specifies the username for an 802.1X credentials profile. |

# dot1x critical (global configuration)

To configure the IEEE 802.1X critical authentication parameters, use the **dot1x critical** command in global configuration mode.

**dot1x critical** {**eapol**| **recovery delay** *milliseconds*}

## Syntax Description

| | |
|---|---|
| **eapol** | Specifies that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. |
| **recovery delay** *milliseconds* | Specifies the recovery delay period that the switch waits to reinitialize a critical port when an unavailable RADIUS server becomes available; valid values are from 1 to 10000, in milliseconds. |

## Command Default

The default settings are as follows:

- **eapol** --Disabled
- *milliseconds* --1000 milliseconds

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |
| 12.2(33)SXI | The **recovery delay** keyword was replaced by the **authentication critical recovery delay** command. |

## Examples

This example shows how to specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port:

```
Switch(config)# dot1x critical eapol
```

This example shows how to set the recovery delay period that the switch waits to reinitialize a critical port when an unavailable RADIUS server becomes available:

```
Switch(config)# dot1x critical recovery delay 1500
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x critical (interface configuration)** | Enables 802.1X critical authentication on an interface. |

# dot1x critical (interface configuration)

To enable 802.1X critical authentication, and optionally, 802.1X critical authentication recovery and authentication, on an interface, use the **dot1x critical** command in interface configuration mode. To disable 802.1X critical authentication, and optionally, 802.1X critical authentication recovery and authentication, use the **no** form of this command.

**dot1x critical** [**recovery action reinitialize**]

**no dot1x critical** [**recovery action reinitialize**]

## Syntax Description

| | |
|---|---|
| **recovery action reinitialize** | (Optional) Enables 802.1X critical authentication recovery and specifies that the port is authenticated when an authentication server is available. |

## Command Default

The 802.1X critical authentication is enabled on an interface.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |

## Examples

This example shows how to enable 802.1X critical authentication on an interface:

```
Router(config-if)# dot1x critical
```

This example shows how to enable 802.1X critical authentication recovery and authenticate the port when an authentication server is available:

```
Router(config-if)# dot1x critical recovery action reinitialize
```

This example shows how to disable 802.1X critical authentication on an interface:

```
Router(config-if)# no
 dot1x critical
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x critical (global configuration)** | Configures the 802.1X critical authentication parameters. |

# dot1x default

To reset the global 802.1X authentication parameters to their default values as specified in the latest IEEE 802.1X standard, use the **dot1x default** command in global configuration or interface configuration mode.

**dot1x default**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The default values are as follows:

- The per-interface 802.1X protocol enable state is disabled (force-authorized).
- The number of seconds between reauthentication attempts is 3600 seconds.
- The quiet period is 60 seconds.
- The retransmission time is 30 seconds.
- The maximum retransmission number is 2 times.
- The multiple host support is disabled.
- The client timeout period is 30 seconds.
- The authentication server timeout period is 30 seconds.

## Command Modes

Global configuration (config) Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.1(6)EA2 | This command was introduced. |
| 12.2(15)ZJ | This command was implemented on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |
| 12.2(14)SX | This command was implemented on the Supervisor Engine 720 in Cisco IOS Release 12.2(14)SX. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |
| 12.2(17d)SXB | This command was implemented on the Supervisor Engine 2 in Cisco IOS Release 12.2(17d)SXB. |
| 12.4(6)T | Interface configuration was added as a configuration mode for this command. |

| Release | Modification |
|---|---|
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Routers (ISRs) only. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |

## Usage Guidelines

The IEEE 802.1x standard defines a client-server-based access control and authentication protocol that restricts unauthorized devices from connecting to a LAN through publicly accessible ports. 802.1x controls network access by creating two distinct virtual access points at each port. One access point is an uncontrolled port; the other is a controlled port. All traffic through the single port is available to both access points. 802.1x authenticates each user device that is connected to a switch port and assigns the port to a VLAN before making available any services that are offered by the switch or the LAN. Until the device is authenticated, 802.1x access control allows only Extensible Authentication Protocol (EAP) over LAN (EAPOL) traffic through the port to which the device is connected. After authentication is successful, normal traffic can pass through the port.

Use the **show dot1x**command to verify your current 802.1X settings.

**Cisco IOS Release 12.4(4)XC**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time; that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

## Examples

The following example shows how to reset the global 802.1X parameters:

```
Router(config)# dot1x default
```

The following example show how to reset the global 802.1X parameters on FastEthernet interface 0:

```
Router(config)# interface FastEthernet0
Router(config-if)# dot1x default
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x critical (global configuration)** | Configures the 802.1X critical authentication parameters. |
| **dot1x critical (interface configuration)** | Enables 802.1X critical authentication on an interface. |
| **dot1x max-req** | Sets the maximum number of times that the device sends an EAP request/identity frame to a client (assuming that a response is not received) before restarting the authentication process. |
| **dot1x re-authentication (EtherSwitch)** | Enables periodic reauthentication of the client for the Ethernet switch network module. |

| Command | Description |
|---|---|
| **dot1x timeout (EtherSwitch)** | Sets retry timeouts for the Ethernet switch network module. |
| **show dot1x** | Displays 802.1X information. |
| **show dot1x (EtherSwitch)** | Displays the 802.1X statistics, administrative status, and operational status for the device or for the specified interface. |

# dot1x guest-vlan

To specify an active VLAN as an IEEE 802.1x guest VLAN, use the **dot1x guest-vlan**command in interface configuration mode. To return to the default setting, use the **no** form of this command.

**dot1x guest-vlan vlan-id**

**no dot1x guest-vlan**

## Syntax Description

| | |
|---|---|
| *vlan-id* | Specify an active VLAN as an IEEE 802.1x guest VLAN. The range is 1 to 4094. |

## Command Default

No guest VLAN is configured.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.1(14)EA1 | This command was introduced. |
| 12.2(25)SE | This command was modified to change the default guest VLAN behavior. |
| 12.4(11)T | This command was integrated into Cisco IOS Release 12.4(11)T. |
| 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. |
| 15.3(1)S | This command was integrated into Cisco IOS Release 15.3(1)S. |

## Usage Guidelines

You can configure a guest VLAN on a static-access port.

For each IEEE 802.1x port, you can configure a guest VLAN to provide limited services to clients (a device or workstation connected to the switch) not running IEEE 802.1x authentication. These users might be upgrading their systems for IEEE 802.1x authentication, and some hosts, such as Windows 98 systems, might not be IEEE 802.1x capable.

When you enable a guest VLAN on an IEEE 802.1x port, the software assigns clients to a guest VLAN when it does not receive a response to its Extensible Authentication Protocol over LAN (EAPOL) request/identity frame or when EAPOL packets are not sent by the client.

With Cisco IOS Release 12.4(11)T and later, the switch port maintains the EAPOL packet history. If another EAPOL packet is detected on the interface during the lifetime of the link, the guest VLAN feature is disabled.

If the port is already in the guest VLAN state, the port returns to the unauthorized state, and authentication restarts. The EAPOL history is reset upon loss of link.

Any number of non-IEEE 802.1x-capable clients are allowed access when the switch port is moved to the guest VLAN. If an IEEE 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the RADIUS-configured or user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on IEEE 802.1x switch ports in single-host or multi-host mode.

You can configure any active VLAN except a Remote Switched Port Analyzer (RSPAN) VLAN or a voice VLAN as an IEEE 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

After you configure a guest VLAN for an IEEE 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the IEEE 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. You should decrease the settings for the IEEE 802.1x authentication process using the **dot1x max-reauth-req**and **dot1x timeout tx-period** interface configuration commands. The amount of decrease depends on the connected IEEE 802.1x client type.

## Examples

This example shows how to specify VLAN 5 as an IEEE 802.1x guest VLAN:

```
Switch(config-if)# dot1x guest-vlan 5
```

This example shows how to set 3 as the quiet time on the switch, to set 15 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request, and to enable VLAN 2 as an IEEE 802.1x guest VLAN when an IEEE 802.1x port is connected to a DHCP client:

```
Switch(config-if)# dot1x timeout max-reauth-req 3
Switch(config-if)# dot1x timeout tx-period 15
Switch(config-if)# dot1x guest-vlan 2
```

You can display the IEEE 802.1x administrative and operational status for the device or for the specified interface by entering the **show dot1x interface***interface-id* ] privileged EXEC command.

## Related Commands

| Command | Description |
|---|---|
| **dot1x max-reauth-req** | Specifies the number of times that the switch retransmits an EAP-request/identity frame to the client before restarting the authentication process. |
| **dot1x timeout** | Sets authentication retry timeouts. |
| **show dot1x** | Displays details for an identity profile. |

# dot1x guest-vlan supplicant

To allow the 802.1x-capable supplicants to enter the guest VLAN, use the **dot1x guest-vlan supplicant**command in global configuration mode. To prevent the 802.1x-capable supplicants from entering the guest VLAN, use the **no** form of this command.

**dot1x guest-vlan supplicant**

**no dot1x guest-vlan supplicant**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The 802.1x-capable supplicants are prevented from entering the guest VLAN.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |

## Examples

This example shows how to allow the 802.1x-capable supplicants to enter the guest VLAN:

```
Router(config)# dot1x guest-vlan supplicant
```

This example shows how to prevent the 802.1x-capable supplicants from entering the guest VLAN:

```
Router(config)# no dot1x guest-vlan supplicant
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x critical (global configuration)** | Configures the 802.1X critical authentication parameters. |
| **dot1x critical (interface configuration)** | Enables 802.1X critical authentication on an interface. |

# dot1x initialize

**Note** Effective with Cisco IOS Release 12.2(33)SXI, the **dot1x initialize** command is replaced by the **clear authentication session** command. See the **clear authentication session** command for more information.

To initialize 802.1X clients on all 802.1X-enabled interfaces, use the **dot1x initialize** command in privileged EXEC mode. This command does not have a **no** form.

**dot1x initialize** [**interface** *interface-name*]

## Syntax Description

| | |
|---|---|
| **interface** *interface-name* | (Optional) Specifies an interface to be initialized. If this keyword is not entered, all interfaces are initialized. |

## Command Default

State machines are not enabled.

## Command Modes

Privileged EXEC

## Command History

| Release | Modification |
|---|---|
| 12.1(14)EA1 | This command was introduced. |
| 12.3(2)XA | This command was integrated into Cisco IOS Release 12.3(2)XA. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |

## Usage Guidelines

Use this command to initialize the 802.1X state machines and to set up a fresh environment for authentication. After you enter this command, the port status becomes unauthorized.

## Examples

The following example shows how to manually initialize a port:

```
Router# dot1x initialize interface gigabitethernet2/0/2
```

You can verify the unauthorized port status by entering the **show dot1x** [**interface** *interface-name*] command.

## Related Commands

| Command | Description |
|---|---|
| show dot1x | Displays details for an identity profile. |

# dot1x mac-auth-bypass

To enable a switch to authorize clients based on the client MAC address, use the **dot1x mac-auth-bypass**command in interface configuration mode. To disable MAC authentication bypass, use the **no** form of this command.

**dot1x mac-auth-bypass [eap]**

**no dot1x mac-auth-bypass**

## Syntax Description

| | |
|---|---|
| **eap** | (Optional) Configures the switch to use Extensible Authentication Protocol (EAP) for authorization. |

## Command Default

MAC authentication bypass is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was introduced. |
| 15.1(4)M | This command was integrated into Cisco IOS Release 15.1(4)M. |

## Usage Guidelin

**Note** To use MAC authentication bypass on a routed port, ensure that MAC address learning is enabled on the port.

When the MAC authentication bypass feature is enabled on an 802.1X port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. If authorization fails, the switch assigns the port to the guest VLAN if a VLAN is configured.

## Examples

This example shows how to enable MAC authentication bypass:

```
Router(config)# interface fastethernet 5/1
Router(config-if)# dot1x mac-auth-bypass
```

This example shows how to configure the switch to use EAP for authorization:

```
Router(config)# interface fastethernet 5/1
Router(config-if)# dot1x mac-auth-bypass eap
```

This example shows how to disable MAC authentication bypass:

```
Router(config)# interface fastethernet 5/1
Router(config-if)# no dot1x mac-auth-bypass
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x critical (global configuration)** | Configures the 802.1X critical authentication parameters. |
| **dot1x critical (interface configuration)** | Enables 802.1X critical authentication on an interface. |

# dot1x max-reauth-req

To set the maximum number of times the authenticator sends an Extensible Authentication Protocol (EAP) request/identity frame (assuming that no response is received) to the client , use the **dot1x max-reauth-req**command in interface configuration mode. To set the maximum number of times to the default setting of 2, use the **no** form of this command.

**dot1x max-reauth-req** *number*

**no dot1x max-reauth-req**

## Syntax Description

| | |
|---|---|
| *number* | Maximum number of times. The range is 1 through 10. The default is 2. |

## Command Default

The command default is 2.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(18)SE | This command was introduced. |
| 12.2(25)SEC | The *number* argument was added. |
| 12.4(6)T | This command was integrated into Cisco IOS Release 12.4(6)T. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Routers (ISRs) only. |

## Usage Guidelines

You should change the default value of this command only to adjust for unusual circumstances, such as unreliable links or specific behavioral problems with certain clients and authentication servers.

**Cisco IOS Release 12.4(4)XC**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time, that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

**Verifying Settings**

You can verify your settings by entering the show dot1x [interface interface-id] command.

## Examples

The following example shows how to set 4 as the number of times that the authentication process is restarted before changing to the unauthorized state:

```
Router(config-if)# dot1x max-reauth-req 4
```

## Examples

The following example shows Layer 3 802.1X support on a switched virtual interface (using a Cisco 870 ISR):

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
 dot1x reauthentication
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x max-req** | Sets the maximum number of times that a device can send an EAP request/identity frame to a client (assuming that a response is not received) before restarting the authentication process . |
| **dot1x timeout tx-period** | Sets the number of seconds that the switch waits for a response to an EAP request or identity frame from the client before resending the request. |
| **show dot1x** | Displays IEEE 802.1X status for the specified port. |

# dot1x max-req

To set the maximum number of times that a networking device or Ethernet switch network module can send an Extensible Authentication Protocol (EAP) request/identity frame to a client (assuming that a response is not received) before restarting the authentication process, use the **dot1x max-req**command in interface configuration or global configuration mode. To set the number of times to the default setting of 2, use the **no** form of this command.

**dot1x max-req** *retry-number*

**no dot1x max-req**

## Syntax Description

| | |
|---|---|
| retry-number | Maximum number of retries. The value is from 1 through 10. The default value is 2. The value is applicable to all EAP packets except for Request ID. |

## Command Default

The default number of retries is 2.

## Command Modes

Interface configuration (config-if) Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.1(6)EA2 | This command was introduced on the Cisco Ethernet switch network module. |
| 12.2(14)SX | This command was implemented on the Supervisor Engine 720 in Cisco IOS Release 12.2(14)SX. |
| 12.2(15)ZJ | This command was implemented on the Cisco Ethernet switch network module on the following platforms in Cisco IOS Release 12.2(15)ZJ: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series. |
| 12.1(11)AX | This command was integrated into Cisco IOS Release 12.1(11)AX. |
| 12.1(14)EA1 | This command was integrated into Cisco IOS Release 12.1(14)EA1 and the configuration mode was changed to interface configuration mode except on the EtherSwitch network module. |
| 12.3(2)XA | This command was integrated into Cisco IOS Release 12.3(2)XA and implemented on the following router platforms: Cisco 806, Cisco 831, Cisco 836, Cisco 837, Cisco 1701, Cisco 1710, Cisco 1721, Cisco 1751-V, and Cisco 1760. |

| Release | Modification |
|---|---|
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T and implemented on the following router platforms: Cisco 1751, Cisco 2610XM, Cisco 2611XM, Cisco 2620XM, Cisco 2621XM, Cisco 2650XM, Cisco 2651XM, Cisco 2691, Cisco 3640, Cisco 3640A, and Cisco 3660. |
| 12.2(17d)SXB | This command was implemented on the Supervisor Engine 2 in Cisco IOS Release 12.2(17d)SXB. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Routers (ISRs) only. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.(33)SXH. |

## Usage Guidelines

The IEEE 802.1x standard defines a client-server-based access control and authentication protocol that restricts unauthorized devices from connecting to a LAN through publicly accessible ports. 802.1x controls network access by creating two distinct virtual access points at each port. One access point is an uncontrolled port; the other is a controlled port. All traffic through the single port is available to both access points. 802.1x authenticates each user device that is connected to a switch port and assigns the port to a VLAN before making available any services that are offered by the switch or the LAN. Until the device is authenticated, 802.1x access control allows only Extensible Authentication Protocol (EAP) over LAN (EAPOL) traffic through the port to which the device is connected. After authentication is successful, normal traffic can pass through the port.

**Note** You should change the default value of this command only to adjust for unusual circumstances, such as unreliable links or specific behavioral problems with certain clients and authentication servers.

### Cisco IOS Release 12.4(4)XC

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time, that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

## Examples

The following example shows that the maximum number of times that the networking device will send an EAP request or identity message to the client PC is 6:

```
Router(config) configure terminal
Router(config)# interface ethernet 0
Router(config-if)# dot1x max-req 6
```

The following example shows how to set the number of times that a switch sends an EAP request or identity frame to 5 before restarting the authentication process:

```
Router(config-if)# dot1x max-req 5
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x port-control** | Enables manual control of the authorization state of a controlled port. |
| **dot1x re-authentication** | Globally enables periodic reauthentication of the client PCs on the 802.1X interface. |
| **dot1x reauthentication (EtherSwitch)** | Enables periodic reauthentication of the Ethernet switch network module client on the 802.1X interface. |
| **dot1x timeout** | Sets retry timeouts. |
| **dot1x timeout (EtherSwitch)** | Sets retry timeouts for the Ethernet switch network module. |
| **show dot1x** | Displays details for an identity profile. |
| **show dot1x (EtherSwitch)** | Displays the 802.1X statistics, administrative status, and operational status for the device or for the specified interface. |

# dot1x multiple-hosts

**Note** This command was replaced by the **dot1x host-mode** command effective with Cisco IOS Release 12.1(14)EA1 and Release 12.4(6)T.

To allow multiple hosts (clients) on an 802.1X-authorized switch port that has the **dot1x port-control** interface configuration command set to **auto**, use the **dot1x multiple-hosts** command in interface configuration mode. To return to the default setting, use the **no** form of this command.

**dot1x multiple-hosts**

**no dot1x multiple-hosts**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Multiple hosts are disabled.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.1(6)EA2 | This command was introduced. |
| 12.2(15)ZJ | This command was implemented on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |
| 12.1(14)EA1 | This command was replaced by the **dot1x host-mode** command in Cisco IOS Release 12.1(14)EA1. |
| 12.4(6)T | This command was replaced by the **dot1x host-mode** command on the T-train. |

## Usage Guidelines

This command is supported only on switch ports.

This command enables you to attach multiple clients to a single 802.1X-enabled port. In this mode, only one of the attached hosts must be successfully authorized for all hosts to be granted network access. If the port becomes unauthorized (reauthentication fails or an Extensible Authentication Protocol over LAN [EAPOL]-logoff message is received), all attached clients are denied access to the network.

Use the **show dot1x**(EtherSwitch)privileged EXEC command with the **interface** keyword to verify your current 802.1X multiple host settings.

## Examples

The following example shows how to enable 802.1X on Fast Ethernet interface 0/1 and to allow multiple hosts:

```
Router(config)# interface fastethernet0/1
Router(config-if)# dot1x port-control auto
Router(config-if)# dot1x multiple-hosts
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x default** | Enables manual control of the authorization state of the port. |
| **show dot1x (EtherSwitch)** | Displays the 802.1X statistics, administrative status, and operational status for the device or for the specified interface. |

# dot1x pae

To set the Port Access Entity (PAE) type, use the **dot1x pae** command in interface configuration mode. To disable the PAE type that was set, use the **no** form of this command.

**dot1x pae** [**supplicant**| **authenticator**| **both**]

**no dot1x pae** [**supplicant**| **authenticator**| **both**]

## Syntax Description

| | |
|---|---|
| **supplicant** | (Optional) The interface acts only as a supplicant and will not respond to messages that are meant for an authenticator. |
| **authenticator** | (Optional) The interface acts only as an authenticator and will not respond to any messages meant for a supplicant. |
| **both** | (Optional) The interface behaves both as a supplicant and as an authenticator and thus will respond to all dot1x messages. |

## Command Default

PAE type is not set.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.3(11)T | This command was introduced. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Routers (ISRs) only. |
| 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 the **dot1x system-auth-control**command has not been configured, the **supplicant** keyword will be the only keyword available for use with this command. (That is, if the **dot1x system-auth-control**command has not been configured, you cannot configure the interface as an authenticator.)

### Cisco IOS Release 12.4(4)XC

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time, that is, if it is configured on Layer2, it cannot also be configured on Layer 3 and vice versa.

## Examples

The following example shows that the interface has been set to act as a supplicant:

```
Router (config)# interface Ethernet1
Router (config-if)# dot1x pae supplicant
```

## Examples

The following example shows Layer 3 802.1X support on a switched virtual interface (using a Cisco 870 ISR):

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
```

dot1x reauthentication

## Related Commands

| Command | Description |
|---|---|
| **dot1x system-auth-control** | Enables 802.1X SystemAuthControl (port-based authentication). |
| **interface** | Configures an interface type. |

# dot1x port-control

**Note** Effective with Cisco IOS Release 12.2(33)SXI, the **dot1x port-control** command is replaced by the **authentication port-control** command. See the **authentication port-control** command for more information.

To enable manual control of the authorization state of a controlled port, use the **dot1x port-control** command in interface configuration mode. To disable the port-control value, use the **no** form of this command.

**dot1x port-control** {**auto**| **force-authorized**| **force-unauthorized**}

**no dot1x port-control**

## Syntax Description

| | |
|---|---|
| **auto** | Enables 802.1X port-based authentication and causes the port to begin in the unauthorized state, allowing only Extensible Authentication Protocol over LAN (EAPOL) frames to be sent and received through the port. |
| **force-authorized** | Disables 802.1X on the interface and causes the port to change to the authorized state without any authentication exchange required. The port transmits and receives normal traffic without 802.1X-based authentication of the client. The **force-authorized** keyword is the default. |
| **force-unauthorized** | Denies all access through this interface by forcing the port to change to the unauthorized state, ignoring all attempts by the client to authenticate. |

## Command Default

The default is force-authorized.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.1(6)EA2 | This command was introduced for the Cisco Ethernet switch network module. |
| 12.1(11)AX | This command was integrated into Cisco IOS Release 12.1(11)AX. |
| 12.2(14)SX | Support for this command was introduced on the Supervisor Engine 720. |

| Release | Modification |
|---|---|
| 12.2(15)ZJ | This command was implemented on the following platforms for the Cisco Ethernet switch network module: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series. |
| 12.3(2)XA | This command was introduced on the following Cisco Switches: Cisco 806, Cisco 831, Cisco 836, Cisco 837, Cisco 1701, Cisco 1710, Cisco 1721, Cisco 1751-V, and Cisco 1760. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. Switch support was added for the following platforms: Cisco 1751, Cisco 2610XM, Cisco 2611XM, Cisco 2620XM, Cisco 2621XM, Cisco 2650XM, Cisco 2651XM, Cisco 2691, Cisco 3640, Cisco 3640A, and Cisco 3660. |
| 12.2(17d)SXB | Support for this command on the Supervisor Engine 2 was added for Cisco IOS Release 12.2(17d)SXB. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Switchs (ISRs) only. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXI | This command was replaced by the **authentication port-control** command. |

## Usage Guidelines

**For Ethernet Switch Network Modules**

The following guidelines apply to Ethernet switch network modules:

- The 802.1X protocol is supported on Layer 2 static-access ports.
- You can use the **auto** keyword only if the port is not configured as one of these types:
  - Trunk port--If you try to enable 802.1X on a trunk port, an error message appears, and 802.1X is not enabled. If you try to change the mode of an 802.1X-enabled port to trunk, the port mode is not changed.
  - EtherChannel port--Before enabling 802.1X on the port, you must first remove it from the EtherChannel. If you try to enable 802.1X on an EtherChannel or on an active port in an EtherChannel, an error appears, and 802.1X is not enabled. If you enable 802.1X on a not-yet active port of an EtherChannel, the port does not join the EtherChannel.
  - Switch Port Analyzer (SPAN) destination port--You can enable 802.1X on a port that is a SPAN destination port; however, 802.1X is disabled until the port is removed as a SPAN destination. You can enable 802.1X on a SPAN source port.

To globally disable 802.1X on the device, you must disable it on each port. There is no global configuration command for this task.

**For Cisco IOS Release 12.4(4)XC**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time; that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

**Verifying Settings**

You can verify your settings by entering the **show dot1x** ommand and checking the Status column in the 802.1X Port Summary section of the display. An enabled status means that the port-control value is set to auto or to force-unauthorized.

## Examples

The following example shows that the authentication status of the client PC will be determined by the authentication process:

```
Switch(config)# configure terminal
Switch(config)# interface ethernet 0
Switch(config-if)# dot1x port-control auto
```

## Examples

The following example shows Layer 3 802.1X support on a switched virtual interface (using a Cisco 870 ISR):

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
```

dot1x reauthentication

## Related Commands

| Command | Description |
|---|---|
| **dot1x max-req** | Sets the maximum number of times that a switch or Ethernet switch network module can send an EAP request/identity frame to a client (assuming that a response is not received) before restarting the authentication process. |
| **dot1x re-authentication** | Globally enables periodic reauthentication of the client on the 802.1X interface. |
| **dot1x reauthentication (EtherSwitch)** | Enables periodic reauthentication of the client on the 802.1X interface. |

| Command | Description |
|---|---|
| **dot1x timeout** | Sets retry timeouts. |
| **dot1x timeout (EtherSwitch)** | Sets retry timeouts for the Ethernet switch network module. |
| **show dot1x** | Displays details for an identity profile. |
| **show dot1x (EtherSwitch)** | Displays the 802.1X statistics, administrative status, and operational status for the switch or for the specified interface. |

# dot1x re-authenticate (privileged EXEC)

**Note** Effective with Cisco IOS Release 12.2(33)SXI, the **dot1x re-authenticate** command is replaced by the **clear authentication session** command. See the **clear authentication session** command for more information.

To manually initiate a reauthentication of the specified 802.1X-enabled ports, use the **dot1x re-authenticate** command in privileged EXEC mode.

**dot1x re-authenticate** [**interface** *interface-name interface-number*]

## Syntax Description

| | |
|---|---|
| **interface** *interface-name interface-number* | (Optional) Interface on which reauthentication is to be initiated. |

## Command Default

There is no default setting.

## Command Modes

Privileged EXEC

## Command History

| Release | Modification |
|---|---|
| 12.1(11)AX | This command was introduced. |
| 12.3(2)XA | This command was integrated into Cisco IOS Release12.3(2)XA. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Routers (ISRs) only. |

## Usage Guidelines

You can use this command to reauthenticate a client without having to wait for the configured number of seconds between reauthentication attempts (re-authperiod) and automatic reauthentication.

**Cisco IOS Release 12.4(4)XC**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time, that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

**Examples**

The following example shows how to manually reauthenticate the device that is connected to a port:

```
Router# dot1x re-authenticate interface gigabitethernet2/0/1
```

**Examples**

The following example shows Layer 3 802.1X support on a switched virtual interface (using a Cisco 870 ISR):

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
```

dot1x reauthentication

**Related Commands**

| Command | Description |
|---|---|
| **dot1x reauthentication** | Globally enables periodic reauthentication of the client PCs on the 802.1X interface. |
| dot1x timeout | Sets retry timeouts. |

# dot1x reauthentication

**Note** Effective with Cisco IOS Release 12.2(33)SXI, the **dot1x reauthentication** command is replaced by the **authentication periodic**command. See the **authentication periodic** command for more information.

To enable periodic reauthentication of the client PCs on the 802.1X interface, use the **dot1x reauthentication** command in interface configuration mode. To disable periodic reauthentication, use the **no** form of this command.

**dot1x reauthentication**

**no dot1x reauthentication**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Periodic reauthentication is not set.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(14)SX | This command was introduced on the Supervisor Engine 720. |
| 12.3(2)XA | This command was integrated into Cisco IOS Release 12.3(2)XA. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |
| 12.2(17d)SXB | This command was implemented on the Supervisor Engine 2 in Cisco IOS Release 12.2(17d)SXB. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Routers (ISRs) only. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXI | This command was replaced by the **authentication periodic** command. |

## Usage Guidelines

The reauthentication period can be set using the **dot1x timeout** command.

**Cisco IOS Release 12.4(4)XC**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time; that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

## Examples

The following example shows that reauthentication has been enabled and the reauthentication period as been set for 1800 seconds:

```
Router(config)# configure terminal
Router(config)# interface ethernet 0
Router(config-if)# dot1x reauthentication
Router(config-if)# dot1x timeout reauth-period 1800
```

## Examples

The following example shows Layer 3 802.1X support on a switched virtual interface using a Cisco 870 ISR:

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
```

dot1x reauthentication

## Examples

The following example shows how to enable periodic reauthentication of the client:

```
Router(config-if)# dot1x reauthentication
Router(config-if)#
```

The following example shows how to disable periodic reauthentication of the client:

```
Router(config-if)# no dot1x reauthentication
Router(config-if)#
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x max-req** | Sets the maximum number of times that a router can send an EAP request/identity frame to a client PC (assuming that a response is not received) before concluding that the client PC does not support 802.1X. |
| **dot1x port-control** | Sets an 802.1X port control value. |

| Command | Description |
|---|---|
| **dot1x timeout** | Sets retry timeouts. |
| **show dot1x** | Displays 802.1X information. |

# dot1x re-authentication (EtherSwitch)

To enable periodic reauthentication of the client for an Ethernet switch network module, use the **dot1x re-authentication** command in global configuration mode. To disable periodic reauthentication, use the **no** form of this command.

**dot1x re-authentication**

**no dot1x re-authentication**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Periodic reauthentication is disabled.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.1(6)EA2 | This command was introduced. |
| 12.2(15)ZJ | This command was implemented on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |

## Usage Guidelines

You configure the amount of time between periodic reauthentication attempts by using the **dot1x timeout re-authperiod** global configuration command.

## Examples

The following example shows how to disable periodic reauthentication of the client:

```
Router(config)# no dot1x re-authentication
```

The following example shows how to enable periodic reauthentication and set the number of seconds between reauthentication attempts to 4000 seconds:

```
Router(config)# dot1x re-authentication
Router(config)# dot1x timeout re-authperiod 4000
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x timeout (EtherSwitch)** | Sets retry timeouts for the Ethernet switch network module. |

| Command | Description |
|---|---|
| **show dot1x (EtherSwitch)** | Displays the 802.1X statistics, administrative status, and operational status for the device or for the specified interface. |

# dot1x system-auth-control

To globally enable 802.1X SystemAuthControl (port-based authentication), use the **dot1x system-auth-control**command in global configuration mode. To disable SystemAuthControl, use the **no** form of this command.

**dot1x system-auth-control**

**no dot1x system-auth-control**

## Syntax Description

This command has no arguments or keywords.

## Command Default

System authentication is disabled by default. If this command is disabled, all ports behave as if they are force authorized.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.3(2)XA | This command was introduced. |
| 12.2(14)SX | This command was implemented on the Supervisor Engine 720. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |
| 12.2(17d)SXB | Support for this command on the Supervisor Engine 2 was extended to 12.2(17d)SXB. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |

## Usage Guidelines

The IEEE 802.1x standard defines a client-server-based access control and authentication protocol that restricts unauthorized devices from connecting to a LAN through publicly accessible ports. 802.1x controls network access by creating two distinct virtual access points at each port. One access point is an uncontrolled port; the other is a controlled port. All traffic through the single port is available to both access points. 802.1x authenticates each user device that is connected to a switch port and assigns the port to a VLAN before making available any services that are offered by the switch or the LAN. Until the device is authenticated, 802.1x access control allows only Extensible Authentication Protocol (EAP) over LAN (EAPOL) traffic through the port to which the device is connected. After authentication is successful, normal traffic can pass through the port.

The **no** form of the command removes any 802.1X-related configurations.

**Catalyst 6500 Series Switch and Cisco 7600 Series**

You must enable Authentication, Authorization, and Accounting (AAA) and specify the authentication method list before enabling 802.1X. A method list describes the sequence and authentication methods to be queried to authenticate a user.

## Examples

The following example shows how to enable SystemAuthControl:

```
Router(config)# dot1x system-auth-control
```

## Related Commands

| Command | Description |
|---|---|
| **aaa authentication dot1x** | Specifies one or more AAA methods for use on interfaces running IEEE 802.1X. |
| **aaa new-model** | Enables the AAA access-control model. |
| **debug dot1x** | Displays 802.1X debugging information. |
| **description** | Specifies a description for an 802.1X profile. |
| **device** | Statically authorizes or rejects individual devices. |
| **dot1x initialize** | Initializes 802.1X state machines on all 802.1X-enabled interfaces. |
| **dot1x max-req** | Sets the maximum number of times that a router or Ethernet switch network module can send an EAP request/identity frame to a client (assuming that a response is not received) before restarting the authentication process. |
| **dot1x port-control** | Enables manual control of the authorized state of a controlled port. |
| **dot1x re-authenticate** | Manually initiates a reauthentication of the specified 802.1X-enabled ports. |
| **dot1x reauthentication** | Globally enables periodic reauthentication of the client PCs on the 802.1X interface. |
| **dot1x timeout** | Sets retry timeouts. |
| **identity profile** | Creates an identity profile and enters identity profile configuration mode. |
| **show dot1x** | Displays details and statistics for an identity profile. |
| **template** | Specifies a virtual template from which commands may be cloned. |

# dot1x timeout

To configure the value for retry timeouts, use the **dot1x timeout** command in global configuration or interface configuration mode. To return to the default value for retry timeouts to, use the **no** form of this command.

### All Platforms Except the Cisco 7600 Series Switch

**dot1x timeout** {**auth-period** *seconds*| **held-period** *seconds*| **quiet-period** *seconds*| **ratelimit-period** *seconds*| **reauth-period** {*seconds*| **server**}| **server-timeout** *seconds*| **start-period** *seconds*| **supp-timeout** *seconds*| **tx-period** *seconds*}

**no dot1x timeout** {**auth-period** *seconds*| **held-period** *seconds*| **quiet-period** *seconds*| **ratelimit-period** *seconds*| **reauth-period** {*seconds*| **server**}| **server-timeout** *seconds*| **start-period** *seconds*| **supp-timeout** *seconds*| **tx-period** *seconds*}

### Cisco 7600 Series Switch

**dot1x timeout** {**reauth-period** *seconds*| **quiet-period** *seconds*| **tx-period** *seconds*| **supp-timeout** *seconds*| **server-timeout** *seconds*}

**no dot1x timeout** {**reauth-period**| **quiet-period**| **tx-period**| **supp-timeout**| **server-timeout**}

## Syntax Description

| | |
|---|---|
| **auth-period** *seconds* | Configures the time, in seconds, the supplicant (client) waits for a response from an authenticator (for packets other than Extensible Authentication Protocol over LAN [EAPOL]-Start) before timing out.<br>• The range is from 1 to 65535. The default is 30. |
| **held-period** *seconds* | Configures the time, in seconds for which a supplicant will stay in the HELD state (that is, the length of time it will wait before trying to send the credentials again after a failed attempt).<br>• The range is from 1 to 65535. The default is 60. |
| **quiet-period** *seconds* | Configures the time, in seconds, that the authenticator (server) remains quiet (in the HELD state) following a failed authentication exchange before trying to reauthenticate the client.<br>• For all platforms except the Cisco 7600 series Switch, the range is from 1 to 65535. The default is 120.<br>• For the Cisco 7600 series Switch, the range is from 0 to 65535. The default is 60. |

| | |
|---|---|
| **ratelimit-period** *seconds* | Throttles the EAP-START packets that are sent from misbehaving client PCs (for example, PCs that send EAP-START packets that result in the wasting of switch processing power).<br>• The authenticator ignores EAPOL-Start packets from clients that have successfully authenticated for the rate-limit period duration.<br>• The range is from 1 to 65535. By default, rate limiting is disabled. |
| **reauth-period** {*seconds* \| **server**} | Configures the time, in seconds, after which an automatic reauthentication should be initiated.<br>• The **server**keyword indicates that the reauthentication period value for the client should be obtained from the authentication, authorization, and accounting (AAA) server as the Session-Timeout (RADIUS Attribute 27) value. If the **server** keyword is used, the action upon reauthentication is also decided by the server and sent as the Termination-Action (RADIUS Attribute 29) value. The termination action could be either "terminate" or "reauthenticate." If the **server** keyword is not used, the termination action is always "reauthenticate."<br>• For all platforms except the Cisco 7600 series switch, the range is from 1 to 65535. The default is 3600.<br>• For the Cisco 7600 series switch, the range is from 1 to 4294967295. The default is 3600. See the "Usage Guidelines" section for additional information.<br>**Note** Effective with Cisco IOS Release 12.2(33)SXI, this phrase is replaced by the **authentication timer reauthenticate**command. See the **authentication timer reauthenticate**command for more information. |

| | |
|---|---|
| **server-timeout** *seconds* | Configures the interval, in seconds, between two successive EAPOL-Start frames when they are being retransmitted.<br>• For all platforms except the Cisco 7600 series switch, the range is from 1 to 65535. The default is 30.<br>• For the Cisco 7600 series switch, the range is from 30 to 65535. The default is 30.<br>If the server does not send a response to an 802.1X packet within the specified period, the packet is sent again. |
| **start-period** *seconds* | Configures the interval, in seconds, between two successive EAPOL-Start frames when they are being retransmitted.<br>• The value is from 1 to 65535. The default is 30. |
| **supp-timeout** *seconds* | Sets the authenticator-to-supplicant retransmission time for all EAP messages other than EAP Request ID.<br>• For all platforms except the Cisco 7600 series Switch, the range is from 1 to 65535. The default is 30.<br>• For the Cisco 7600 series Switch, the range is from 30 to 65535. The default is 30. |
| **tx-period** *seconds* | Configures the number of seconds between retransmission of EAP request ID packets (assuming that no response is received) to the client.<br>• For all platforms except the Cisco 7600 series switch, the range is from 1 to 65535. The default is 30.<br>• For the Cisco 7600 series switch, the range is from 30 to 65535. The default is 30.<br>• If an 802.1X packet is sent to the supplicant and the supplicant does not send a response after the retry period, the packet will be sent again. |

**Command Default** Periodic reauthentication and periodic rate-limiting are not done.

**Command Modes**

Global configuration Interface configuration

**Cisco 7600 Switch**

Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 12.2(14)SX | This command was introduced on the Supervisor Engine 720. |
| 12.3(2)XA | This command was integrated into Cisco IOS Release 12.3(2)XA. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |
| 12.2(18)SE | Ranges for the **server-timeout**, **supp-timeout**, and **tx-period** keywords were changed. |
| 12.2(17d)SXB | Support for this command on the Supervisor Engine 2 was added for Cisco IOS Release 12.2(17d)SXB. |
| 12.3(11)T | The **auth-period**, **held-period**, and **start-period** keywords were added. |
| 12.2(25)SEC | The range for the **tx-period** keyword was changed, and the **reauth-period** and **server-timeout** keywords were added. |
| 12.1(11)AX | This command was introduced. |
| 12.1(14)EA1 | The **supp-timeout** and **server-timeout** keywords were added. The configuration mode for the command was changed to interface configuration mode. |
| 12.4(6)T | The **supp-timeout** keyword was added, and this command was integrated into Cisco IOS Release 12.4(6)T. |
| 12.4(4)XC | This command was integrated into Cisco IOS Release 12.4(4)XC for Cisco 870 Integrated Services Switchs (ISRs) only. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXI | The **reauth-period** keyword was replaced by the **authentication timer reauthenticate**command. |

**Usage Guidelines**

For Cisco IOS Release 12.4(4)XC, on Cisco 870 ISRs only, this command can be configured on Layer 2 (for switch ports) and Layer 3 (for switched virtual interfaces). However, the command can function at only one layer at a time; that is, if it is configured on Layer 2, it cannot also be configured on Layer 3 and vice versa.

**Cisco 7600 Switch**

You must enable periodic reauthentication before you enter the **dot1x timeout reauth-period** command. Enter the **dot1x reauthentication** command to enable periodic reauthentication. The **dot1x timeout reauth-period** command affects the behavior of the system only if periodic reauthentication is enabled.

## Examples

The following example shows that various 802.1X retransmission and timeout periods have been set:

```
Switch(config)# configure terminal
Switch(config)# interface ethernet 0
Switch(config-if)# dot1x port-control auto
Switch(config-if)# dot1x reauthentication
Switch(config-if)# dot1x timeout auth-period 2000
Switch(config-if)# dot1x timeout held-period 2400
Switch(config-if)# dot1x timeout reauth-period 1800
Switch(config-if)# dot1x timeout quiet-period 600
Switch(config-if)# dot1x timeout start-period 90
Switch(config-if)# dot1x timeout supp-timeout 300
Switch(config-if)# dot1x timeout tx-period 60
Switch(config-if)# dot1x timeout server-timeout 60
```

The following example shows how to return to the default reauthorization period:

```
Switch(config-if)# no dot1x timeout reauth-period
```

## Examples

The following example shows how to set 802.1X retransmission and timeout periods on the Cisco 7600 Switch:

```
Switch(config-if)# dot1x timeout reauth-period 4000
Switch(config-if)# dot1x timeout tx-period 60
Switch(config-if)# dot1x timeout supp-timeout 25
Switch(config-if)# dot1x timeout server-timeout 25
```

## Examples

The following example shows Layer 3 802.1X support on a switched virtual interface (using a Cisco 870 ISR):

```
interface FastEthernet0
 description  switchport connect to a client
!
interface FastEthernet1
 description  switchport connect to a client
!
interface FastEthernet2
 description  switchport connect to a client
!
interface FastEthernet3
 description  switchport connect to a client
!
interface FastEthernet4
 description  Connect to the public network
!
interface Vlan1
 description  Apply 802.1x functionality on SVI
 dot1x pae authenticator
 dot1x port-control auto
```

dot1x reauthentication

## Related Commands

| Command | Description |
|---|---|
| **dot1x max-req** | Sets the maximum number of times that a switch or Ethernet switch module can send an EAP request/identity frame to a client (assuming that a response is not received) before restarting the authentication process. |
| **dot1x port-control** | Sets an 802.1X port control value. |
| **dot1x re-authentication** | Globally enables periodic reauthentication of the client PCs on the 802.1X interface. |
| **show dot1x** | Displays 802.1X information. |

# dot1x timeout (EtherSwitch)

To set the number of retry seconds between 802.1X authentication exchanges when an Ethernet switch network module is installed in the router, use the **dot1x timeout**command in global configuration mode. To return to the default setting, use the **no** form of this command.

**dot1x timeout** {**quiet-period** *seconds*| **re-authperiod** *seconds*| **tx-period** *seconds*}

**no dot1x timeout** {**quiet-period** *seconds*| **re-authperiod** *seconds*| **tx-period** *seconds*}

## Syntax Description

| | |
|---|---|
| **quiet-period** *seconds* | Specifies the time in seconds that the Ethernet switch network module remains in the quiet state following a failed authentication exchange with the client. The range is from 0 to 65535 seconds. The default is 60 seconds. |
| **re-authperiod** *seconds* | Specifies the number of seconds between reauthentication attempts. The range is from 1 to 4294967295. The default is 3660 seconds. |
| **tx-period** *seconds* | Time in seconds that the switch should wait for a response to an EAP-request/identity frame from the client before retransmitting the request. The range is from 1 to 65535 seconds. The default is 30 seconds. |

## Command Default

**quiet-period** : 60 seconds **re-authperiod**: 3660 seconds**tx-period**: 30 seconds

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.1(6)EA2 | This command was introduced. |
| 12.2(15)ZJ | This command was implemented on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T on the following platforms: Cisco 2600 series, Cisco 3600 series, and Cisco 3700 series routers. |

## Usage Guidelines

You should change the default values of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients or authentication servers.

### quiet-period Keyword

During the quiet period, the Ethernet switch network module does not accept or initiate any authentication requests. If you want to provide a faster response time to the user, enter a smaller number than the default.

### re-authperiod Keyword

The **re-authperiod** keyword affects the behavior of the the Ethernet switch network module only if you have enabled periodic reauthentication by using the **dot1x re-authentication** global configuration command.

## Examples

The following example shows how to set the quiet time on the switch to 30 seconds:

```
Router(config)# dot1x timeout quiet-period 30
```

The following example shows how to enable periodic reauthentication and set the number of seconds between reauthentication attempts to 4000 seconds:

```
Router(config)# dot1x re-authentication
Router(config)# dot1x timeout re-authperiod 4000
```

The following example shows how to set 60 seconds as the amount of time that the switch waits for a response to an EAP-request/identity frame from the client before retransmitting the request:

```
Router(config)# dot1x timeout tx-period 60
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x max-req** | Sets the maximum number of times that the device sends an EAP-request/identity frame before restarting the authentication process. |
| **dot1x re-authentication (EtherSwitch)** | Enables periodic reauthentication of the client for the Ethernet switch network module. |
| **show dot1x (EtherSwitch)** | Displays the 802.1X statistics, administrative status, and operational status for the device or for the specified interface. |

# E

# enable password

To set a local password to control access to various privilege levels, use the **enable password** command in global configuration mode. To remove the password requirement, use the **no**form of this command.

**enable password** [**level** *level*] {*password*| [ *encryption-type* ] *encrypted-password*}

**no enable password** [**level** *level*]

## Syntax Description

| | |
|---|---|
| **level** *level* | (Optional) Level for which the password applies. You can specify up to 16 privilege levels, using numbers 0 through 15. Level 1 is normal EXEC-mode user privileges. If this argument is not specified in the command or the **no** form of the command, the privilege level defaults to 15 (traditional enable privileges). |
| *password* | Password users type to enter enable mode. |
| *encryption-type* | (Optional) Cisco-proprietary algorithm used to encrypt the password. Currently the only encryption type available is 5. If you specify *encryption-type*, the next argument you supply must be an encrypted password (a password already encrypted by a Cisco router). |
| *encrypted-password* | Encrypted password you enter, copied from another router configuration. |

## Command Default

No password is defined. The default is level 15.

## 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 Guidelin

**Caution** If neither the enable password command nor the enable secret command is configured, and if there is a line password configured for the console, the console line password will serve as the enable password for all VTY (Telnet and Secure Shell [SSH]) sessions.

Use this command with the **level** option to define a password for a specific privilege level. After you specify the level and the password, give the password to the users who need to access this level. Use the **privilege level** configuration command to specify commands accessible at various levels.

You will not ordinarily enter an encryption type. Typically you enter an encryption type only if you copy and paste into this command a password that has already been encrypted by a Cisco router.

**Caution** If you specify an encryption type and then enter a clear text password, you will not be able to reenter enable mode. You cannot recover a lost password that has been encrypted by any method.

If the **service password-encryption** command is set, the encrypted form of the password you create with the **enable password** command is displayed when a **more nvram:startup-config**command is entered.

You can enable or disable password encryption with the **service password-encryption**command.

An enable password is defined as follows:

- Must contain from 1 to 25 uppercase and lowercase alphanumeric characters.
- Can have leading spaces, but they are ignored. However, intermediate and trailing spaces are recognized.
- Can contain the question mark (?) character if you precede the question mark with the key combinationCrtl-v when you create the password; for example, to create the password *abc?123,* do the following:
  - Enter **abc**.
  - Type **Crtl-v**.
  - Enter **?123**.

When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.

## Examples

The following example enables the password " pswd2" for privilege level 2:

```
enable password level 2 pswd2
```

The following example sets the encrypted password "$1$i5Rkls3LoyxzS8t9", which has been copied from a router configuration file, for privilege level 2 using encryption type 7:

```
enable password level 2 5 $1$i5Rkls3LoyxzS8t9
```

## Related Commands

| Command | Description |
|---|---|
| **disable** | Exits privileged EXEC mode and returns to user EXEC mode. |
| **enable** | Enters privileged EXEC mode. |
| **enable secret** | Specifies an additional layer of security over the **enable password** command. |
| **privilege** | Configures a new privilege level for users and associate commands with that privilege level. |
| **service password-encryption** | Encrypts passwords. |
| **show privilege** | Displays your current level of privilege. |

# enable secret

To specify an additional layer of security over the **enable password** command, use the **enable secret** command in global configuration mode. To turn off the **enable secret** function, use the **no** form of this command.

**enable secret** [**level** *level*] {**[0]** *unencrypted-password*| *encryption-type encrypted-password*}

**no enable secret** [**level** *level*] [*encryption-type encrypted-password*]

## Syntax Description

| | |
|---|---|
| **level** *level* | (Optional) Specifies the level for which the password applies. You can specify up to 15 privilege levels, using numerals 1 through 15. Level 1 is normal EXEC-mode user privileges. If the *level* argument is not specified in the command or in the **no** form of the command, the privilege level defaults to 15 (traditional enable privileges). |
| **0** | (Optional) Specifies an unencrypted clear-text password. The password is converted to a Secure Hash Algorithm (SHA) 256 secret and gets stored in the router. |
| *unencrypted-password* | Password for users to enter enable mode. This password should be different from the password created with the **enable password** command. |
| *encryption-type* | Cisco-proprietary algorithm used to encrypt the password. The encryption types available for this command are 4 and 5.<br>• **4 —**Specifies an SHA-256 encrypted secret string. The SHA256 secret string is copied from the router configuration.<br>• **5 —**Specifies a message digest alogrithm 5 (MD5) encrypted secret. |
| *encrypted-password* | Encrypted password that is copied from another router configuration. |

## Command Default

No password is defined.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 11.0 | 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. |
| 15.0(1)S | This command was integrated into Cisco IOS Release 15.0(1)S. Support for the encryption type **4** was added. |
| Cisco IOS XE Release 3.1S | This command was integrated into Cisco IOS XE Release 3.1S. Support for the encryption type **4** was added. |
| 15.1(4)M | This command was modified. Support for the encryption type **4** was added. |
| Cisco IOS Release 3.3SG | This command was modified. Support for the encryption type **5** was removed. |
| 15.1(1)SY | This command was integrated into Cisco IOS Release 15.1(1)SY. |
| Cisco IOS XE Release 3.8S | This command was modified. The warning message for removal of support for the encryption type **5** was modified. |

## Usage Guidelin

**Caution** If neither the **enable password** command or the **enable secret** command is configured, and if a line password is configured for the console, the console line password will serve as the enable password for all vty (Telnet and Secure Shell [SSH]) sessions.

Use the **enable secret** command to provide an additional layer of security over the enable password. The **enable secret** command provides better security by storing the enable secret password using a nonreversible cryptographic function. The added layer of security encryption provides is useful in environments where the password crosses the network or is stored on a TFTP server.

Typically you enter an encryption type only when you paste an encrypted password that you copied from a router configuration file into this command.

**Caution** If you specify an encryption type and then enter a clear-text password, you will not be able to reenter enable mode. You cannot recover a lost password that has been encrypted by any method.

If you use the same password for the **enable password** and **enable secret** commands, you receive an error message warning that this practice is not recommended, but the password will be accepted. By using the same password, however, you undermine the additional security the **enable secret** command provides.

**Note** After you set a password using the **enable secret** command, a password set using the **enable password** command works only if the **enable secret** is disabled or an older version of Cisco IOS software is being used, such as when running an older rxboot image. Additionally, you cannot recover a lost password that has been encrypted by any method.

If the **service password-encryption** command is set, the encrypted form of the password you create is displayed when the **more nvram:startup-config** command is entered.

You can enable or disable password encryption with the **service password-encryption** command.

An enable password is defined as follows:

- Must contain 1 to 25 alphanumeric characters, both uppercase and lowercase.
- Can have leading spaces, but they are ignored. However, intermediate and trailing spaces are recognized.
- Can contain the question mark (?) character if you precede the question mark with the key combinationCrtl-v when you create the password; for example, to create the password *abc?123,* do the following:
  - Enter **abc**.
  - Press **Crtl-v**.
  - Enter **?123**.

When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can enter **abc?123** at the password prompt.

**Note** During a downgrade from Cisco IOS XE Release 3.3SG to Cisco IOS XE Release 3.2SG, if a SHA256-encrypted enable password is configured, then the SHA256-encrypted password will be lost without any warning, and the secret password will have to be reconfigured.

## Examples

The following example shows how to specify the password with the **enable secret** command:

```
Device> enable
Device# configure terminal
Device(config)# enable secret password
```

After specifying a password with the **enable secret** command, users must enter this password to gain access. Any passwords set through **enable password** command will no longer work.

```
Password: password
```

The following example shows how to enable the encrypted password "$1$FaD0$Xyti5Rkls3LoyxzS8", which has been copied from a router configuration file, for privilege level 2 using the encryption type 4:

```
Device> enable
Device# configure terminal
Device(config)# enable password level 2 4 $1$FaD0$Xyti5Rkls3LoyxzS8
```

The following example is a sample warning message that is displayed when a user enters the **enable secret 5** *encrypted-password* command:

```
Device(config)# enable secret 5 $1$FaD0$Xyti5Rkls3LoyxzS8

Warning: The CLI will be deprecated soon
'enable secret 5 <password>'
Please move to 'enable secret <password>' CLI
```

## Related Commands

| Command | Description |
|---|---|
| **enable** | Enters privileged EXEC mode. |
| **enable password** | Sets a local password to control access to various privilege levels. |
| **more nvram:startup-config** | Displays the startup configuration file contained in NVRAM or specified by the CONFIG_FILE environment variable. |
| **service password-encryption** | Encrypt passwords. |

# enrollment http-proxy

To access the certification authority (CA) by HTTP through the proxy server, use the **enrollment http-proxy** command in ca-trustpoint configuration mode.

**enrollment http-proxy** *host-name port-num*

## Syntax Description

| | |
|---|---|
| *host-name* | Defines the proxy server used to get the CA. |
| *port-num* | Specifies the port number used to access the CA. |

## Command Default

If this command is not enabled, the CA will not be accessed via HTTP.

## Command Modes

Ca-trustpoint configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(8)T | This command was introduced. |
| 12.2(18)SXD | This command was integrated into Cisco IOS Release 12.2(18)SXD. |

## Usage Guidelines

The **enrollment http-proxy** command must be used in conjunction with the **enrollment** command, which specifies the enrollment parameters for the CA.

## Examples

The following example shows how to access the CA named “ka” by HTTP through the bomborra proxy server:

```
crypto ca trustpoint ka
 enrollment url http://kahului
 enrollment http-proxy bomborra 8080
 crl optional
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ca trustpoint** | Declares the CA that your router should use. |
| **enrollment** | Specifies the enrollment parameters of your CA. |

# enrollment url (ca-profile-enroll)

To specify the URL of the certification authority (CA) server to which to send enrollment requests, use the **enrollment url**command in ca-profile-enroll configuration mode. To delete the enrollment URL from your enrollment profile, use the **no** form of this command.

**enrollment url** *url*

**no enrollment url** *url*

## Syntax Description

| | |
|---|---|
| *url* | URL of the CA server to which your router should send certificate requests.<br>If you are using Simple Certificate Enrollment Protocol (SCEP) for enrollment, the *url* argument must be in the form http://CA_name, where CA_name is the host Domain Name System (DNS) name or IP address of the CA.<br>If you are using TFTP for enrollment, the *url* argument must be in the form tftp://certserver/file_specification. (If the URL does not include a file specification, the fully qualified domain name [FQDN] of the router will be used.) |

## Command Default

Your router does not recognize the CA URL until you specify it using this command.

## Command Modes

Ca-profile-enroll configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(13)ZH | This command was introduced. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |

## Usage Guidelines

This command allows the user to specify a different URL or a different method for authenticating a certificate and enrolling a certificate; for example, manual authentication and TFTP enrollment.

## Examples

The following example shows how to enable certificate enrollment via HTTP for the profile name “E” :

```
crypto pki trustpoint Entrust
  enrollment profile E
  serial
```

```
crypto pki profile enrollment E
 authentication url  http://entrust:81
 authentication command  GET /certs/cacert.der
 enrollment url  http://entrust:81/cda-cgi/clientcgi.exe
 enrollment command  POST reference_number=$P2&authcode=$P1
&retrievedAs=rawDER&action=getServerCert&pkcs10Request=$REQ
 parameter 1 value aaaa-bbbb-cccc
 parameter 2 value 5001
```

## Related Commands

| Command | Description |
|---|---|
| **crypto pki profile enrollment** | Defines an enrollment profile. |

# F through H

# hostname (IKEv2 keyring)

To specify the hostname for the peer in the Internet Key Exchange Version 2 (IKEv2) keyring, use the **hostname** command IKEv2 keyring peer configuration mode. To remove the hostname, use the **no** form of this command.

**hostname** *name*

**no hostname**

## Syntax Description

| | |
|---|---|
| *name* | Name for the peer. |

## Command Default

The hostname is not specified.

## Command Modes

IKEv2 keyring peer configuration (config-ikev2-keyring-peer)

## Command History

| Release | Modification |
|---|---|
| 15.1(1)T | This command was introduced. |
| 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

When configuring the IKEv2 keyring, use this command to identify the peer using hostname, which is:

- Independent of the IKEv2 identity.
- Available on an IKEv2 initiator only.
- Provided by IPsec to IKEv2 as part of a security association setup request to identify the peer.
- Used to identify the peer only with crypto maps and not with tunnel protection.

## Examples

The following example shows how to configure the hostname for a peer when configuring an IKEv2 keyring:

```
Router(config)# crypto ikev2 keyring keyring-1
Router(config-ikev2-keyring)# peer peer1
Router(config-ikev2-keyring-peer)# description peer1
Router(config-ikev2-keyring-peer)# hostname peer1.example.com
```

## Related Commands

| Command | Description |
|---|---|
| address (ikev2 keyring) | Specifies the IPv4 address or the range of the peers in IKEv2 key. |
| **crypto ikev2 keyring** | Defines an IKEv2 keyring. |
| **description (ikev2 keyring)** | Describes an IKEv2 peer or a peer group for the IKEv2 keyring. |
| **identity (ikev2 keyring)** | Identifies the peer with IKEv2 types of identity. |
| **peer** | Defines a peer or a peer group for the keyring. |
| pre-shared-key **(ikev2 keyring)** | Defines a preshared key for the IKEv2 peer. |

# icmp idle-timeout through ip http ezvpn

# identity profile

To create an identity profile and to enter identity profile configuration mode, use the **identity profile**command in global configuration mode. To disable an identity profile, use the **no** form of this command.

**identity profile** {**default**| **dot1x**| **eapoudp**| **auth-proxy**}

**no identity profile** {**default**| **dot1x**| **eapoudp**| **auth-proxy**}

## Syntax Description

| | |
|---|---|
| **default** | Service type is default. |
| **dot1x** | Service type for 802.1X. |
| **eapoudp** | Service type for Extensible Authentication Protocol over UDP (EAPoUDP). |
| **auth-proxy** | Service type for authentication proxy. |

## Command Default

An identity profile is not created.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.3(2)XA | This command was introduced. |
| 12.3(4)T | This command was integrated into Cisco IOS Release 12.3(4)T. |
| 12.3(8)T | The **eapoudp** keyword was added. |
| 12.4(6)T | The **dot1x** keyword was removed. |
| 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 **identity profile** command and **default** keyword allow you to configure static MAC addresses of a client computer that does not support 802.1X and to authorize or unauthorize them statically. After you have issued the **identity profile** command and **default** keyword and the router is in identity profile configuration mode,

you can specify the configuration of a template that can be used to create the virtual access interface to which unauthenticated supplicants (client computers) will be mapped.

The **identity profile** command and the **dot1x** keyword are used by the supplicant and authenticator. Using the **dot1x** keyword, you can set the username, password, or other identity-related information for an 802.1X authentication.

Using the **identity profile** command and the **eapoudp** keyword, you can statically authenticate or unauthenticate a device either on the basis of the device IP address or MAC address or on the type, and the corresponding network access policy can be specified using the **identity policy** command.

## Examples

The following example shows that an identity profile and its description have been specified:

```
Router (config)# identity profile default
Router (config-identity-prof)# description description_entered_here
```

The following example shows that an EAPoUDP identity profile has been created:

```
Router (config)# identity policy eapoudp
```

## Related Commands

| Command | Description |
|---|---|
| **debug dot1x** | Displays 802.1X debugging information. |
| **description** | Specifies a description for an 802.1X profile. |
| **device** | Statically authorizes or rejects individual devices. |
| **dot1x initialize** | Initializes 802.1X state machines on all 802.1X-enabled interfaces. |
| **dot1x max-req** | Sets the maximum number of times that a router can send an EAP request/identity frame to a client PC. |
| **dot1x max-start** | Sets the maximum number of times the authenticator sends an EAP request/identity frame (assuming that no response is received) to the client. |
| **dot1x pae** | Sets the PAE type during 802.1X authentication. |
| **dot1x port-control** | Enables manual control of the authorization state of a controlled port. |
| **dot1x re-authenticate** | Manually initiates a reauthtication of the specified 802.1X-enabled ports. |
| **dot1x re-authentication** | Globally enables periodic reauthentication of the client PCs on the 802.1X interface. |
| **dot1x system-auth-control** | Enables 802.1X SystemAuthControl (port-based authentication). |

| Command | Description |
|---|---|
| **dot1x timeout** | Sets retry timeouts. |
| **identity policy** | Creates an identity policy. |
| **show dot1x** | Displays details for an identity profile. |
| **template (identity profile)** | Specifies a virtual template from which commands may be cloned. |

# ip access-group

To apply an IP access list or object group access control list (OGACL) to an interface or a service policy map, use the **ip access-group** command in the appropriate configuration mode. To remove an IP access list or OGACL, use the **no** form of this command.

**ip access-group** {*access-list-name*| *access-list-number*} {**in**| **out**}

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

## Syntax Description

| | |
|---|---|
| *access-list-name* | Name of the existing IP access list or OGACL as specified by an **ip access-list** command. |
| *access-list-number* | Number of the existing access list.<br>• Integer from 1 to 199 for a standard or extended IP access list.<br>• Integer from 1300 to 2699 for a standard or extended IP expanded access list. |
| **in** | Filters on inbound packets. |
| **out** | Filters on outbound packets. |

## Command Default

An access list is not applied.

## Command Modes

Interface configuration (config-if) Service policy-map configuration (config-service-policymap)

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 11.2 | The *access-list-name* argument was added. |
| 12.2(28)SB | This command was made available in service policy-map configuration mode. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | The *access-list-name* keyword was modified to accept the name of an OGACL. |

| Release | Modification |
|---|---|
| Cisco IOS XE 3.3S | This command was integrated into Cisco IOS XE Release 3.3S. |
| 15.2(02)SA | This command was implemented on the Cisco ME 2600X Series Ethernet Access Switches. |

## Usage Guidelines

If the specified access list does not exist, all packets are passed (no warning message is issued).

**Applying Access Lists to Interfaces**

Acc ess lists or OGACLs are applied on either outbound or inbound interfaces. For standard inbound access lists, after an interface receives a packet, the Cisco IOS software checks the source address of the packet against the access list. For extended access lists or OGACLs, the networking device also checks the destination access list or OGACL. If the access list or OGACL permits the address, the software continues to process the packet. If the access list or OGACL rejects the address, the software discards the packet and returns an Internet Control Management Protocol (ICMP) host unreachable message.

For standard outbound access lists, after a device receives and routes a packet to a controlled interface, the software checks the source address of the packet against the access list. For extended access lists or OGACLs, the networking device also checks the destination access list or OGACL. If the access list or OGACL permits the address, the software sends the packet. If the access list or OGACL rejects the address, the software discards the packet and returns an ICMP host unreachable message.

When you enable outbound access lists or OGACLs, you automatically disable autonomous switching for that interface. When you enable inbound access lists or OGACLs on any CBus or CxBus interface, you automatically disable autonomous switching for all interfaces (with one exception--a Storage Services Enabler (SSE) configured with simple access lists can still switch packets, on output only).

**Applying Access Lists or OGACLs to Service Policy Maps**

You can use the **ip access-group** command to configure Intelligent Services Gateway (ISG) per-subscriber firewalls. Per-subscriber firewalls are Cisco IOS IP access lists or OGACLs that are used to prevent subscribers, services, and pass-through traffic from accessing specific IP addresses and ports.

ACLs and OGACLs can be configured in user profiles or service profiles on an authentication, authorization, and accounting (AAA) server or in service policy maps on an ISG. OGACLS or numbered or named IP access lists can be configured on the ISG, or the ACL or OGACL statements can be included in the profile configuration.

When an ACL or OGACL is added to a service, all subscribers of that service are prevented from accessing the specified IP address, subnet mask, and port combinations through the service.

## Examples

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

```
Router> enable
Router# configure terminal
Router(config)# interface ethernet 0
Router(config-if)# ip access-group 101 out
```

## Related Commands

| Command | Description |
|---|---|
| **deny** | Sets conditions in a named IP access list or OGACL that will deny packets. |
| **ip access-list** | Defines an IP access list or OGACL by name or number. |
| **object-group network** | Defines network object groups for use in OGACLs. |
| **object-group service** | Defines service object groups for use in OGACLs. |
| **permit** | Sets conditions in a named IP access list or OGACL that will permit packets. |
| **show ip access-list** | Displays the contents of IP access lists or OGACLs. |
| **show object-group** | Displays information about object groups that are configured. |

# ip access-list

To define an IP access list or object-group access control list (ACL) by name or number or to enable filtering for packets with IP helper-address destinations, use the **ip access-list** command in global configuration mode. To remove the IP access list or object-group ACL or to disable filtering for packets with IP helper-address destinations, use the **no** form of this command.

**ip access-list** {{**standard**| **extended**} {*access-list-name*| *access-list-number*}| **helper egress check**}

**no ip access-list** {{**standard**| **extended**} {*access-list-name*| *access-list-number*}| **helper egress check**}

## Syntax Description

| | |
|---|---|
| **standard** | Specifies a standard IP access list. |
| **extended** | Specifies an extended IP access list. Required for object-group ACLs. |
| *access-list-name* | Name of the IP access list or object-group ACL. Names cannot contain a space or quotation mark, and must begin with an alphabetic character to prevent ambiguity with numbered access lists. |
| *access-list-number* | Number of the access list.<br>• A standard IP access list is in the ranges 1-99 or 1300-1999.<br>• An extended IP access list is in the ranges 100-199 or 2000-2699. |
| **helper egress check** | Enables permit or deny matching capability for an outbound access list that is applied to an interface, for traffic that is relayed via the IP helper feature to a destination server address. |

## Command Default

No IP access list or object-group ACL is defined, and outbound ACLs do not match and filter IP helper relayed traffic.

## Command Modes

Global configuration (config)

## 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. |

| Release | Modification |
|---|---|
| 12.2(33)SXH | This command was integrated into Cisco IOS Release 12.2(33)SXH. |
| 12.4(20)T | This command was modified. Object-group ACLs are now accepted when the **deny** and **permit** commands are used in standard IP access-list configuration mode or extended IP access-list configuration mode. |
| Cisco IOS XE Release 3.2S | This command was implemented on Cisco ASR 1000 series routers. |
| 15.0(1)M5 | This command was modified. The **helper**, **egress**, and **check** keywords were added. |
| 15.1(1)SY | This command was modified. The **helper**, **egress**, and **check** keywords were added. |
| 15.1(3)T3 | This command was modified. The **helper**, **egress**, and **check** keywords were added. |
| 15.1(2)SNG | This command was implemented on the Cisco ASR 901 Series Aggregation Services Routers. |

## Usage Guidelines

Use this command to configure a named or numbered IP access list or an object-group ACL. This command places the router in access-list configuration mode, where you must define the denied or permitted access conditions by using the **deny**and **permit** commands.

Specifying the **standard** or **extended** keyword with the **ip access-list** command determines the prompt that appears when you enter access-list configuration mode. You must use the **extended** keyword when defining object-group ACLs.

You can create object groups and IP access lists or object-group ACLs independently, which means that you can use object-group names that do not yet exist.

Named access lists are not compatible with Cisco IOS software releases prior to Release 11.2.

Use the **ip access-group** command to apply the access list to an interface.

The **ip access-list helper egress check** command enables outbound ACL matching for permit or deny capability on packets with IP helper-address destinations. When you use an outbound extended ACL with this command, you can permit or deny IP helper relayed traffic based on source or destination User Datagram Protocol (UDP) ports. The **ip access-list helper egress check** command is disabled by default; outbound ACLs will not match and filter IP helper relayed traffic.

## Examples

The following example defines a standard access list named Internetfilter:

```
Router> enable
Router# configure terminal
Router(config)# ip access-list standard Internetfilter
Router(config-std-nacl)# permit 192.168.255.0 0.0.0.255
Router(config-std-nacl)# permit 10.88.0.0 0.0.255.255
Router(config-std-nacl)# permit 10.0.0.0 0.255.255.255
```

The following example shows how to create an object-group ACL that permits packets from the users in my_network_object_group if the protocol ports match the ports specified in my_service_object_group:

```
Router> enable
Router# configure terminal
Router(config)# ip access-list extended my_ogacl_policy
Router(config-ext-nacl)# permit tcp object-group my_network_object_group portgroup
 my_service_object_group any
Router(config-ext-nacl)# deny tcp any any
```

The following example shows how to enable outbound ACL filtering on packets with helper-address destinations:

```
Router> enable
Router# configure terminal
Router(config)# ip access-list helper egress check
```

## Related Commands

| Command | Description |
|---|---|
| **deny** | Sets conditions in a named IP access list or in an object-group ACL that will deny packets. |
| **ip access-group** | Applies an ACL or an object-group ACL to an interface or a service policy map. |
| **object-group network** | Defines network object groups for use in object-group ACLs. |
| **object-group service** | Defines service object groups for use in object-group ACLs. |
| **permit** | Sets conditions in a named IP access list or in an object-group ACL that will permit packets. |
| **show ip access-list** | Displays the contents of IP access lists or object-group ACLs. |
| **show object-group** | Displays information about object groups that are configured. |

# ip access-list resequence

To apply sequence numbers to the access list entries in an access list, use the **ip access-list resequence** command in global configuration mode.

**ip access-list resequence** *access-list-name* **starting-sequence-number** *increment*

## Syntax Description

| | |
|---|---|
| *access-list-name* | Name of the access list. Names cannot contain a space or quotation mark. |
| *starting-sequence-number* | Access list entries will be resequenced using this initial value. The default value is 10. The range of possible sequence numbers is 1 through 2147483647. |
| *increment* | The number by which the sequence numbers change. The default value is 10. For example, if the increment value is 5 and the beginning sequence number is 20, the subsequent sequence numbers are 25, 30, 35, 40, and so on. |

## Command Default

Disabled

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(14)S | This command was introduced. |
| 12.2(15)T | This command was integrated into Cisco IOS Release 12.2(15)T. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

This command allows the **permit** and **deny** entries of a specified access list to be resequenced with an initial sequence number value determined by the *starting-sequence-number* argument, and continuing in increments determined by the *increment* argument. If the highest sequence number exceeds the maximum possible sequence number, then no sequencing occurs.

For backward compatibility with previous releases, if entries with no sequence numbers are applied, the first entry is assigned a sequence number of 10, and successive entries are incremented by 10. The maximum sequence number is 2147483647. If the generated sequence number exceeds this maximum number, the following message is displayed:

```
Exceeded maximum sequence number.
```

If the user enters an entry without a sequence number, it is assigned a sequence number that is 10 greater than the last sequence number in that access list and is placed at the end of the list.

If the user enters an entry that matches an already existing entry (except for the sequence number), then no changes are made.

If the user enters a sequence number that is already present, the following error message is generated:

```
Duplicate sequence number.
```

If a new access list is entered from global configuration mode, then sequence numbers for that access list are generated automatically.

Distributed support is provided so that the sequence numbers of entries in the Route Processor (RP) and line card (LC) are in synchronization at all times.

Sequence numbers are not saved in NVRAM. That is, the sequence numbers themselves are not saved. In the event that the system is reloaded, the configured sequence numbers revert to the default sequence starting number and increment.

This command works with named standard and extended IP access lists. Because the name of an access list can be designated as a number, numbers are acceptable as names as long as they are entered in named access list configuration mode.

## Examples

The following example resequences an access list named kmd1. The starting sequence number is 100, and the increment value is 5:

```
ip access-list resequence kmd1 100 5
```

## Related Commands

| Command | Description |
|---|---|
| **deny (IP)** | Sets conditions under which a packet does not pass a named IP access list. |
| **permit (IP)** | Sets conditions under which a packet passes a named IP access list. |

# ip admission

To create a Layer 3 network admission control rule to be applied to the interface, or to create a policy that can be applied on an interface when the authentication, authorization and accounting (AAA) server is unreachable, use the **ip admission**command in interface configuration mode. To create a global policy that can be applied on a network access device, use the **ip admission command with the optional keywords**and argument in global configuration mode. To remove the admission control rule, use the **no** form of this command.

**ip admission** *admission-name* [**event timeout aaa policy identity** *identity-policy-name*]

**no ip admission** *admission-name* [**event timeout aaa policy identity** *identity-policy-name*]

## Syntax Description

| | |
|---|---|
| *admission-name* | Authentication or admission rule name. |
| **event timeout aaa policy identity** | Specifies an authentication policy to be applied when the AAA server is unreachable. |
| *identity-policy-name* | Authentication or admission rule name to be applied when the AAA server is unreachable. |

## Command Default

A network admission control rule is not applied to the interface.

## Command Modes

Interface configuration (config-if) Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.3(8)T | This command was introduced. |
| 12.4(11)T | This command was modified to include the **event timeout aaa policy identity** keywords and the *identity-policy-name* argument. |
| 12.2(33)SXI | This command was integrated into Cisco IOS Release 12.2(33)SXI. |

## Usage Guidelines

The admission rule defines how you apply admission control.

The optional keywords and argument define the network admission policy to be applied to a network access device or an interface when no AAA server is reachable. The command can be used to associate a default identity policy with Extensible Authentication Protocol over User Datagram Protocol (EAPoUDP) sessions.

## Examples

The following example shows how to apply a network admission control rule named "nacrule1" to the interface:

```
Router (config-if)# ip admission nacrule1
```

The following example shows how to apply an identity policy named "example" to the device when the AAA server is unreachable:

```
Router (config)# ip admission nacrule1 event timeout aaa policy identity example
```

## Related Commands

| Command | Description |
|---|---|
| **interface** | Defines an interface. |

# ip admission proxy http

To specify the display of custom authentication proxy web pages during web-based authentication, use the **ip admission proxy http** command in global configuration mode. To specify the use of the default web page, use the **no** form of this command.

**ip admission proxy http** {{**login**| **success**| **failure**| **login expired**} **page file** *device:file-name*| **success redirect** *url*}

**no ip admission proxy http** {{**login**| **success**| **failure**| **login expired**} **page file** *device:file-name*| **success redirect** *url*}

## Syntax Description

| | |
|---|---|
| **login** | Specifies a locally stored web page to be displayed during login. |
| **success** | Specifies a locally stored web page to be displayed when the login is successful. |
| **failure** | Specifies a locally stored web page to be displayed when the login has failed. |
| **login expired** | Specifies a locally stored web page to be displayed when the login has expired. |
| *device* | Specifies a disk or flash memory in the switch memory file system where the custom HTML file is stored. |
| *file-name* | Specifies the name of the custom HTML file to be used in place of the default HTML file for the specified condition. |
| **success redirect** *url* | Specifies an external web page to be displayed when the login is successful. |

## Command Default

The internal default authentication proxy web pages are displayed during web-based authentication.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXI | This command was introduced. |

## Usage Guidelines

When configuring the use of customized authentication proxy web pages, consider the following guidelines:

- To enable the custom web pages feature, you must specify all four custom HTML files. If fewer than four files are specified, the internal default HTML pages will be used.
- The four custom HTML files must be present on the disk or flash of the switch. The maximum size of each HTML file is 8 KB.
- Any images on the custom pages must be located on an accessible HTTP server. An intercept ACL must be configured within the admission rule to allow access to the HTTP server.
- Any external link from a custom page will require configuration of an intercept ACL within the admission rule.
- Any name resolution required for external links or images will require configuration of an intercept ACL within the admission rule to access a valid DNS server.
- If the custom web pages feature is enabled, a configured auth-proxy-banner will not be used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature will not be available.
- Because the custom login page is a public web form, consider the following guidelines for this page:
  - The login form must accept user input for the username and password and must POST the data as uname and pwd.
  - The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.
- When configuring a redirection URL for successful login, consider the following guidelines:
  - If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and will not be available in the CLI. You can perform redirection in the custom login success page.
  - If the redirection URL feature is enabled, a configured auth-proxy-banner will not be used.

## Examples

The following example shows how to configure custom authentication proxy web pages:

```
Router(config)# ip admission proxy http login page file disk1:login.htm
Router(config)# ip admission proxy http success page file disk1:success.htm
Router(config)# ip admission proxy http fail page file disk1:fail.htm
Router(config)# ip admission proxy http login expired page file disk1:expired.htm
```

The following example shows how to verify the configuration of custom authentication proxy web pages:

```
Router# show ip admission configuration
Authentication proxy webpage
 Login page         : disk1:login.htm
 Success page       : disk1:success.htm
 Fail Page          : disk1:fail.htm
 Login expired Page : disk1:expired.htm
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Session ratelimit is 100
Authentication Proxy Watch-list is disabled
```

```
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

The following example shows how to configure a redirection URL for successful login:

```
Router(config)# ip admission proxy http success redirect www.example.com
```

The following example shows how to verify the redirection URL for successful login:

```
Router# show ip admission configuration
Authentication Proxy Banner not configured
Customizable Authentication Proxy webpage not configured
HTTP Authentication success redirect to URL: http://www.example.com
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled
Authentication Proxy Max HTTP process is 7
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

## Related Commands

| Command | Description |
|---|---|
| ip http server ip https server | Enables the HTTP server within the switch. |
| **show ip admission configuration** | Displays the configuration of web-based authentication ip admission. |

# ip device tracking probe

To enable the tracking of device probes, use the **ip device tracking probe** command in configuration mode. To disable device probes, use the **no** form of this command.

**ip device tracking probe** {**count** *count*| **delay** *delay*| **interval** *interval*}

## Syntax Description

| | |
|---|---|
| **count** *count* | Specifies the number of IP tracking probes from 1 to 5. |
| **delay** *delay* | Specifies the delay time of IP tracking probes from 1 to 120 seconds. |
| **interval** *interval* | Specifies the time between IP tracking probes from 30 to 300 minutes. |

## Command Default

Device probe tracking is disabled.

## Command Modes

Config mode (config #)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SXI7 | This command was introduced. |

## Examples

The following example shows how to set the probe count to 5:

```
Router(config)# ip device tracking probe count 5
```

The following example shows how to set the delay time to 60:

```
Router(config)# ip device tracking probe delay 60
```

The following example shows how to set the interval time to 35:

```
Router(config)# ip device tracking probe interval 35
```

## Related Commands

| Command | Description |
|---|---|
| **show ip device tracking** | Displays information about entries in the IP device tracking table. |

# ip inspect through ip security strip

# 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:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

**Note** When using scp, you cannot enter the password into the **copy** command; enter the password when prompted.

## Related Commands

| Command | Description |
|---|---|
| **aaa authentication login** | Sets AAA authentication at login. |
| **aaa authorization** | Sets parameters that restrict user access to a network. |
| **copy** | Copies any file from a source to a destination. |
| **debug ip scp** | Troubleshoots scp authentication problems. |
| **ip ssh port** | Enables secure network access to the tty lines. |
| **username** | Establishes a username-based authentication system. |

# ip source-track through ivrf

# ip ssh

To configure Secure Shell (SSH) control parameters on your router, use the **ip ssh** command in global configuration mode. To restore the default value, use the **no** form of this command.

**ip ssh** [**timeout** *seconds*| **authentication-retries** *integer*]

**no ip ssh** [**timeout** *seconds*| **authentication-retries** *integer*]

## Syntax Description

| | |
|---|---|
| **timeout** | (Optional) The time interval that the router waits for the SSH client to respond.<br>This setting applies to the SSH negotiation phase. Once the EXEC session starts, the standard timeouts configured for the vty apply. By default, there are 5 vtys defined (0-4), therefore 5 terminal sessions are possible. After the SSH executes a shell, the vty timeout starts. The vty timeout defaults to 10 minutes. |
| *seconds* | (Optional) The number of seconds until timeout disconnects, with a maximum of 120 seconds. The default is 120 seconds. |
| **authentication- retries** | (Optional) The number of attempts after which the interface is reset. |
| *integer* | (Optional) The number of retries, with a maximum of 5 authentication retries. The default is 3. |

## Command Default

SSH control parameters are set to default router values.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.0(5)S | This command was introduced. |
| 12.1(1)T | This command was integrated into Cisco IOS Release 12.1(1) T. |
| 12.2(17a)SX | This command was integrated into Cisco IOS Release 12.2(17a)SX. |
| 12.2(33)SRA | This command was integrated into Cisco IOS release 12.(33)SRA. |
| Cisco IOS XE Release 2.4 | This command was implemented on the Cisco ASR 1000 series routers. |

## Usage Guidelines

Before you configure SSH on your router, you must enable the SSH server using the **crypto key generate rsa**command.

## Examples

The following examples configure SSH control parameters on your router:

```
ip ssh timeout 120
ip ssh authentication-retries 3
```

# ip ssh dh min size

To configure the modulus size on the Secure Shell (SSH) server, use the **ip ssh dh min size** command in privileged EXEC mode. To disable the configuration, use the **no** form of this command.

**ip ssh dh min size** [ *number* ]

**no ip ssh dh min size**

## Syntax Description

| | |
|---|---|
| *number* | (Optional) Minimum number of bits in the key size. The default is 1024. |

## Command Default

Bit key support is disabled.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| 12.4(20)T | This command was introduced. |
| 15.1(2)S | This command was integrated into Cisco IOS Release 15.1(2)S. |

## Usage Guidelines

Use the **ip ssh dh min size**command to ensure that the CLI is successfully parsed from either the client side or the server side.

## Examples

The following example shows how to set the minimum modulus size to 2048 bits:

```
Router> enable
Router# ip ssh dh min size 2048
```

## Related Commands

| Command | Description |
|---|---|
| **show ip ssh** | Displays the status of SSH server connections. |

# ip ssh dscp

To specify the IP differentiated services code point (DSCP) value that can be set for a Secure Shell (SSH) configuration, use the **ip ssh dscp**command in global configuration mode. To restore the default value, use the **no** form of this command.

**ip ssh dscp** *number*

**no ip ssh dscp** *number*

## Syntax Description

| | |
|---|---|
| *number* | Value that can be set. The default value is 0 (zero).<br>• *number* --0 through 63. |

## Command Default

The IP DSCP value is not specified.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 12.4(20)S | This command was introduced. |
| 12.2SR | This command is supported in the Cisco IOS Release 12.2SR train. Support in a specific 12.2SR train depends on your feature set, platform, and platform hardware. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX train depends on your feature set, platform, and platform hardware. |
| 12.4(22)T | This command was integrated into Cisco IOS Release 12.4(22)T. |

## Usage Guidelines

IP DSCP values can be configured on both the SSH client and the SSH server for SSH traffic that is generated on either end.

## Examples

The following example shows that the DSCP value is set to 35:

```
Router(config)# ip ssh dscp 35
```

## Related Commands

| Command | Description |
|---|---|
| **ip ssh precedence** | Specifies the IP precedence value that may be set. |

# ip ssh pubkey-chain

To configure Secure Shell RSA (SSH-RSA) keys for user and server authentication on the SSH server, use the **ip ssh pubkey-chain** command in global configuration mode. To remove SSH-RSA keys for user and server authentication on the SSH server, use the **no** form of this command.

**ip ssh pubkey-chain**

**no ip ssh pubkey-chain**

## Syntax Description

This command has no arguments or keywords.

## Command Default

SSH-RSA keys are not configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 15.0(1)M | This command was introduced. |
| 15.1(1)S | This command was integrated into Cisco IOS Release 15.1(1)S. |

## Usage Guidelines

Use the **ip ssh pubkey-chain**command to ensure SSH server and user public key authentication.

## Examples

The following example shows how to enable public key generation:

```
Router(config)# ip ssh pubkey-chain
```

## Related Commands

| Command | Description |
|---|---|
| **ip ssh stricthostkeycheck** | Enables strict host key checking on the SSH server. |

# ip ssh stricthostkeycheck

To enable strict host key checking on the Secure Shell (SSH) server, use the **ip ssh stricthostcheck** command in global configuration mode. To disable strict host key checking, use the **no** form of this command.

**ip ssh stricthostkeycheck**

**no ip ssh stricthostkeycheck**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Strict host key checking on the SSH server is not enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| 15.0(1)M | This command was introduced. |
| 15.1(1)S | This command was integrated into Cisco IOS Release 15.1(1)S. |

## Usage Guidelines

Use the **ip ssh stricthostkeycheck**command to ensure SSH server side strict checking. Configuring the **ip ssh stricthostkeycheck** command authenticates all servers.

**Note** This command is not available on SSH Version 1.

- If the **ip ssh pubkey-chain** command is not configured, the **ip ssh stricthostkeycheck** command will lead to connection failure in SSH Version 2.

## Examples

The following example shows how to enable strict host key checking:

```
Router(config)# ip ssh stricthostkeycheck
```

## Related Commands

| Command | Description |
|---|---|
| **ip ssh pubkey-chain** | Configures SSH-RSA keys for user and server authentication on the SSH server. |

# ip ssh version

To specify the version of Secure Shell (SSH) to be run on a router, use the **ip ssh version**command in global configuration mode. To disable the version of SSH that was configured and to return to compatibility mode, use the **no** form of this command.

**ip ssh version** [**1**| **2**]

**no ip ssh version** [**1**| **2**]

## Syntax Description

| | |
|---|---|
| **1** | (Optional) Router runs only SSH Version 1. |
| **2** | (Optional) Router runs only SSH Version 2. |

## Command Default

If this command is not configured, SSH operates in compatibility mode, that is, Version 1 and Version 2 are both supported.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 12.3(4)T | This command was introduced. |
| 12.3(2)XE | This command was integrated into Cisco IOS Release 12.3(2)XE. |
| 12.2(25)S | This command was integrated into Cisco IOS Release 12.2(25)S. |
| 12.3(7)JA | This command was integrated into Cisco IOS Release 12.3(7)JA. |
| 12.0(32)SY | This command was integrated into Cisco IOS Release 12.0(32)SY. |
| 12.4(20)T | This command was integrated into Cisco IOS Release 12.4(20)T. |
| 15.2(2)SA2 | This command was implemented on the Cisco ME 2600X Series Ethernet Access Switches. |

## Usage Guidelines

You can use this command with the **2** keyword to ensure that your router will not inadvertently establish a weaker SSH Version 1 connection.

## Examples

The following example shows that only SSH Version 1 support is configured:

```
Router (config)# ip ssh version 1
```

The following example shows that only SSH Version 2 is configured:

```
Router (config)# ip ssh version 2
```

The following example shows that SSH Versions 1 and 2 are configured:

```
Router (config)# no ip ssh version
```

## Related Commands

| Command | Description |
|---|---|
| debug ip ssh | Displays debug messages for SSH. |
| disconnect ssh | Terminates a SSH connection on your router. |
| **ip ssh** | Configures SSH control parameters on your router. |
| ip ssh rsa keypair-name | Specifies which RSA key pair to use for a SSH connection. |
| show ip ssh | Displays the SSH connections of your router. |

# ip verify unicast reverse-path

**Note** This command was replaced by the **ip verify unicast source reachable-via**command effective with Cisco IOS Release 12.0(15)S. The **ip verify unicast source reachable-via** command allows for more flexibility and functionality, such as supporting asymmetric routing, and should be used for any Reverse Path Forward implementation. The **ip verify unicast reverse-path** command is still supported.

To enable Unicast Reverse Path Forwarding (Unicast RPF), use the **ip verify unicast reverse-path**command in interface configuration mode. To disable Unicast RPF, use the **no** form of this command.

**ip verify unicast reverse-path** [ *list* ]

**no ip verify unicast reverse-path** [ *list* ]

## Syntax Description

| | |
|---|---|
| *list* | (Optional) Specifies a numbered access control list (ACL) in the following ranges:<br>• 1 to 99 (IP standard access list)<br>• 100 to 199 (IP extended access list)<br>• 1300 to 1999 (IP standard access list, expanded range)<br>• 2000 to 2699 (IP extended access list, expanded range) |

## Command Default

Unicast RPF is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification | | |
|---|---|---|---|
| 11.1(CC)<br>12.0 | This command was introduced. This command was not included in Cisco IOS Release 11.2 or 11.3 | 12.1(2)T | Added ACL support using the *list* argument. Added per-interface statistics on dropped or suppressed packets. |

| Release | Modification |
|---|---|
| 12.0(15)S | The **ip verify unicast source reachable-via** command replaced this command, and the following keywords were added to the **ip verify unicast source reachable-via** command: **allow-default, allow-self-ping**, **rx**, and **any**. |
| 12.1(8a)E | The **ip verify unicast reverse-path**command was integrated into Cisco IOS Release 12.1(8a)E. |
| 12.2(14)S | The **ip verify unicast reverse-path**command was integrated into Cisco IOS Release 12.2(14)S. |
| 12.2(14)SX | The **ip verify unicast reverse-path** command was integrated into Cisco IOS Release 12.2(14)SX. |
| 12.2(33)SRA | The **ip verify unicast reverse-path**command was integrated into Cisco IOS Release 12.2(33)SRA. |

## Usage Guidelines

Use the **ip verify unicast reverse-path interface** command to mitigate problems caused by malformed or forged (spoofed) IP source addresses that are received by a router. Malformed or forged source addresses can indicate denial of service (DoS) attacks on the basis of source IP address spoofing.

When Unicast RPF is enabled on an interface, the router examines all packets that are received on that interface. The router checks to ensure that the source address appears in the Forwarding Information Base (FIB) and that it matches the interface on which the packet was received. This "look backwards" ability is available only when Cisco Express Forwarding is enabled on the router because the lookup relies on the presence of the FIB. Cisco Express Forwarding generates the FIB as part of its operation.

To use Unicast RPF, enable Cisco Express Forwarding switching or distributed Cisco Express Forwarding switching in the router. There is no need to configure the input interface for Cisco Express Forwarding switching. As long as Cisco Express Forwarding is running on the router, individual interfaces can be configured with other switching modes.

**Note** It is very important for Cisco Express Forwarding to be configured globally in the router. Unicast RPF will not work without Cisco Express Forwarding.

**Note** Unicast RPF is an input function and is applied on the interface of a router only in the ingress direction.

The Unicast Reverse Path Forwarding feature checks to determine whether any packet that is received at a router interface arrives on one of the best return paths to the source of the packet. The feature does this by doing a reverse lookup in the Cisco Express Forwarding table. If Unicast RPF does not find a reverse path for the packet, Unicast RPF can drop or forward the packet, depending on whether an ACL is specified in the Unicast Reverse Path Forwarding command. If an ACL is specified in the command, then when (and only when) a packet fails the Unicast RPF check, the ACL is checked to determine whether the packet should be dropped (using a deny statement in the ACL) or forwarded (using a permit statement in the ACL). Whether

a packet is dropped or forwarded, the packet is counted in the global IP traffic statistics for Unicast RPF drops and in the interface statistics for Unicast RPF.

If no ACL is specified in the Unicast Reverse Path Forwarding command, the router drops the forged or malformed packet immediately and no ACL logging occurs. The router and interface Unicast RPF counters are updated.

Unicast RPF events can be logged by specifying the logging option for the ACL entries used by the Unicast Reverse Path Forwarding command. Log information can be used to gather information about the attack, such as source address, time, and so on.

**Where to Use RPF in Your Network**

Unicast RPF may be used on interfaces in which only one path allows packets from valid source networks (networks contained in the FIB). Unicast RPF may also be used in cases for which a router has multiple paths to a given network, as long as the valid networks are switched via the incoming interfaces. Packets for invalid networks will be dropped. For example, routers at the edge of the network of an Internet service provider (ISP) are likely to have symmetrical reverse paths. Unicast RPF may still be applicable in certain multi-homed situations, provided that optional Border Gateway Protocol (BGP) attributes such as weight and local preference are used to achieve symmetric routing.

With Unicast RPF, all equal-cost "best" return paths are considered valid. This means that Unicast RPF works in cases where multiple return paths exist, provided that each path is equal to the others in terms of the routing cost (number of hops, weights, and so on) and as long as the route is in the FIB. Unicast RPF also functions where Enhanced Internet Gateway Routing Protocol (EIGRP) variants are being used and unequal candidate paths back to the source IP address exist.

For example, routers at the edge of the network of an ISP are more likely to have symmetrical reverse paths than routers that are in the core of the ISP network. Routers that are in the core of the ISP network have no guarantee that the best forwarding path out of the router will be the path selected for packets returning to the router. In this scenario, you should use the new form of the command, **ip verify unicast source reachable-via**, if there is a chance of asymmetrical routing.

## Examples

The following example shows that the Unicast Reverse Path Forwarding feature has been enabled on a serial interface:

```
ip cef
! or "ip cef distributed" for RSP+VIP based routers
!
interface serial 5/0/0
 ip verify unicast reverse-path
```

The following example uses a very simple single-homed ISP to demonstrate the concepts of ingress and egress filters used in conjunction with Unicast RPF. The example illustrates an ISP-allocated classless interdomain routing (CIDR) block 192.168.202.128/28 that has both inbound and outbound filters on the upstream interface. Be aware that ISPs are usually not single-homed. Hence, provisions for asymmetrical flows (when outbound traffic goes out one link and returns via a different link) need to be designed into the filters on the border routers of the ISP.

```
ip cef distributed
!
interface Serial 5/0/0
 description Connection to Upstream ISP
 ip address 192.168.200.225 255.255.255.255
 no ip redirects
 no ip directed-broadcast
 no ip proxy-arp
 ip verify unicast reverse-path
 ip access-group 111 in
```

```
 ip access-group 110 out
!
access-list 110 permit ip 192.168.202.128 10.0.0.31 any
access-list 110 deny ip any any log
access-list 111 deny ip host 10.0.0.0 any log
access-list 111 deny ip 172.16.0.0 255.255.255.255 any log
access-list 111 deny ip 10.0.0.0 255.255.255.255 any log
access-list 111 deny ip 172.16.0.0 255.255.255.255 any log
access-list 111 deny ip 192.168.0.0 255.255.255.255 any log
access-list 111 deny ip 209.165.202.129 10.0.0.31 any log
access-list 111 permit ip any any
```

The following example demonstrates the use of ACLs and logging with Unicast RPF. In this example, extended ACL 197 provides entries that deny or permit network traffic for specific address ranges. Unicast RPF is configured on Ethernet interface 0 to check packets arriving at that interface.

For example, packets with a source address of 192.168.201.10 arriving at Ethernet interface 0 are dropped because of the deny statement in ACL 197. In this case, the ACL information is logged (the logging option is turned on for the ACL entry) and dropped packets are counted per-interface and globally. Packets with a source address of 192.168.201.100 arriving at Ethernet interface 0 are forwarded because of the permit statement in ACL 197. ACL information about dropped or suppressed packets is logged (the logging option is turned on for the ACL entry) to the log server.

```
ip cef distributed
!
int eth0/1/1
 ip address 192.168.200.1 255.255.255.255
 ip verify unicast reverse-path 197
!
int eth0/1/2
 ip address 192.168.201.1 255.255.255.255
!
access-list 197 deny   ip 192.168.201.0 10.0.0.63 any log-input
access-list 197 permit ip 192.168.201.64 10.0.0.63 any log-input
access-list 197 deny   ip 192.168.201.128 10.0.0.63 any log-input
access-list 197 permit ip 192.168.201.192 10.0.0.63 any log-input
access-list 197 deny ip host 10.0.0.0 any log-input
access-list 197 deny ip 172.16.0.0 255.255.255.255 any log-input
access-list 197 deny ip 10.0.0.0 255.255.255.255 any log-input
access-list 197 deny ip 172.16.0.0 255.255.255.255 any log-input
access-list 197 deny ip 192.168.0.0 255.255.255.255 any log-input
```

## Related Commands

| Command | Description |
|---|---|
| **ip cef** | Enables Cisco Express Forwarding on the route processor card. |

# ipv6 tacacs source-interface

To specify an interface to use for the source address in TACACS packets, use the **ipv6 tacacs source-interface**command in global configuration mode. To remove the specified interface from the configuration, use the **no** form of this command.

**ipv6 tacacs source-interface** *interface*

**no ipv6 tacacs source-interface** *interface*

## Syntax Description

| | |
|---|---|
| interface | Interface to be used for the source address in TACACS packets. |

## Command Default

No interface is specified.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Release 3.2S | This command was introduced. |

## Usage Guidelines

The **ipv6 tacacs source-interface**command specifies an interface to use for the source address in TACACS packets.

## Examples

The following example shows how to configure the Gigabit Ethernet interface to be used as the source address in TACACS packets:

```
Router(config)# ipv6 tacacs source-interface GigabitEthernet 0/0/0
```

## Related Commands

| Command | Description |
|---|---|
| **tacacs server** | Configures the TACACS+ server for IPv6 or IPv4 and enters TACACS+ server configuration mode. |

# K through L

# key (config-radius-server)

To specify the authentication and encryption key for all RADIUS communications between the router and the RADIUS server, use the **key** command in RADIUS server configuration mode. To remove the configured key, use the **no** form of this command.

**key** {**0** *string*| **7** *string*} *string*

**no key**

## Syntax Description

| | |
|---|---|
| **0**<br>*string* | Specifies that an unencrypted key will follow.<br>The unencrypted (cleartext) shared key. |
| **7**<br>*string* | Specifies that a hidden key will follow.<br>The hidden shared key. |
| *string* | The unencrypted (cleartext) shared key. |

## Command Default

The authentication and encryption key is disabled.

## Command Modes

RADIUS server configuration (config-radius-server)

## Command History

| Release | Modification |
|---|---|
| 15.2(2)T | This command was introduced. |

## Usage Guidelines

After enabling authentication, authorization, and accounting (AAA) authentication with the **aaa new-model** command, you must set the authentication and encryption key using the **radius server key** command.

**Note** Specify a RADIUS key after you issue the **aaa new-model** command.

The key entered must match the key used on the RADIUS server. All leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

## Examples

The following example shows how to specify the host with IP address 192.0.2.2 as the RADIUS server and set rad123 as the encryption key:

```
Device(config)# aaa new-model
```

```
Device(config)# radius server myserver
Device(config-radius-server)# address ipv4 192.0.2.2
Device(config-radius-server)# key rad123
```

The following example shows how to set the authentication and encryption key to anykey. The 7 specifies that a hidden key will follow.

```
Device(config)# aaa new-model
Device(config)# radius server myserver
Device(config-radius-server)# address ipv4 192.0.2.2
Device(config-radius-server)# key 7 anykey
```

After you save your configuration and use the **show running-config** command, an encrypted key will be displayed as follows:

```
Device# show running-config

radius server myserver
  address ipv4 192.0.2.2
  key 7 19283103834782sda
! The leading 7 indicates that the following text is encrypted.
```

## Related Commands

| Command | Description |
|---|---|
| **aaa new-model** | Enables the AAA access control model. |
| **address ipv4** | Configures the IPv4 address for the RADIUS server accounting and authentication parameters. |
| **radius server** | Specifies the name for the RADIUS server configuration and enters RADIUS server configuration mode. |
| **show running-config** | Displays the current configuration of your routing device. |

# key (TACACS+)

To configure the per-server encryption key on the TACACS+ server, use the **key**command in TACACS+ server configuration mode. To remove the per-server encryption key, use the **no** form of this command.

**key** [**0**| **7**] *key-string*

**no key** [**0**| **7**] *key-string*

## Syntax Description

| | |
|---|---|
| **0** | (Optional) Specifies that an unencrypted key will follow. |
| **7** | (Optional) Specifies that a hidden key will follow. |
| key-string | Unencrypted shared key. |

## Command Default

No TACACS+ encryption key is configured.

## Command Modes

TACACS+ server configuration (config-server-tacacs)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Release 3.2S | This command was introduced. |

## Usage Guidelines

The **key**command allows you to configure a per-server encryption key.

## Examples

The following example shows how to specify an unencrypted shared key named key1:

```
Router (config)# tacacs server server1
Router(config-server-tacacs)# key 0 key1
```

## Related Commands

| Command | Description |
|---|---|
| **tacacs server** | Configures the TACACS+ server for IPv6 or IPv4 and enters TACACS+ server configuration mode. |

# key-hash

To specify the Secure Shell (SSH) Rivest, Shamir, and Adleman (RSA) key type and name, use the **key-hash**command in SSH public key configuration mode. To remove the SSH RSA Rivest, Shamir, and Adleman (RSA) public key, use the **no** form of this command.

**key-hash** *key-type key-name*

**no key-hash** [*key-type key-name*]

## Syntax Description

| | |
|---|---|
| *key-type key-name* | The SSH RSA public key type and name. |

## Command Default

SSH key type and name are not specified.

## Command Modes

SSH public key configuration (conf-ssh-pubkey-user)

## Command History

| Release | Modification |
|---|---|
| 12.2(33)SRA | This command was introduced in release earlier than Cisco IOS Release 12.(33)SRA. |

## Usage Guidelines

The key type must be **ssh-rsa** for configuration of private-public key pairs. You can use a hashing software to compute the hash of the public key string or you can copy the hash value from another Cisco IOS router. Using the **key-string** command is the preferred method for entering the public key data for the first time.

## Examples

The following example shows how to specify the SSH key type and name:

```
Router(config)# ip ssh pubkey-chain
Router(conf-ssh-pubkey)# username test
Router(conf-ssh-pubkey-user)# key-hash ssh-rsa key1
Router(conf-ssh-pubkey-user))# exit
Router(config-pubkey)# exit
Router(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **key-string** | Specifies the SSH RSA public key of the remote peer. |

# load-balance (server-group)

To enable RADIUS server load balancing for a named RADIUS server group, use the load-balance command in server group configuration mode. To disable named RADIUS server load balancing, use the **no** form of this command.

**load-balance method least-outstanding** [**batch-size** *number*] **[ignore-preferred-server]**

**no load-balance**

## Syntax Description

| | |
|---|---|
| **method least-outstanding** | Enables least outstanding mode for load balancing. |
| **batch-size** | (Optional) The number of transactions to be assigned per batch. |
| *number* | (Optional) The number of transactions in a batch.<br>• The default is 25.<br>• The range is 1-2147483647.<br>**Note** Batch size may impact throughput and CPU load. It is recommended that the default batch size, 25, be used because it is optimal for high throughput, without adversely impacting CPU load. |
| **ignore-preferred-server** | (Optional) Indicates if a transaction associated with a single authentication, authorization, and accounting (AAA) session should attempt to use the same server or not.<br>• If set, preferred server setting will not be used.<br>• Default is to use the preferred server. |

## Command Default

If this command is not configured, named RADIUS server load balancing will not occur.

## Command Modes

Server group configuration

## Command History

| Release | Modification |
|---|---|
| 12.2(28)SB | This command was introduced. |
| 12.4(11)T | This command was integrated into Cisco IOS Release 12.4(11)T. |

| Release | Modification |
|---|---|
| 12.2(33)SRC | This command was integrated into Cisco IOS Release 12.2(33)SRC. |

**Examples** The following example shows load balancing enabled for a named RADIUS server group. It is shown in three parts: the current configuration of RADIUS command output, debug output, and AAA server status information.

**Examples** The following shows the relevant RADIUS configuration:

```
Router# show running-config
.
.
.
aaa group server radius server-group1
 server 192.0.2.238 auth-port 2095 acct-port 2096
 server 192.0.2.238 auth-port 2015 acct-port 2016
 load-balance method least-outstanding batch-size 5
!
aaa authentication ppp default group server-group1
aaa accounting network default start-stop group server-group1
.
.
.
```

The lines in the current configuration of RADIUS command output above are defined as follows:

- The **aaa group server radius** command shows the configuration of a server group with two member servers.
- The **load-balance** command enables load balancing for the global RADIUS server groups with the batch size specified.
- The **aaa authentication ppp**command authenticates all PPP users using RADIUS.
- The **aaa accounting** command enables the sending of all accounting requests to the AAA server after the client is authenticated and after the disconnect using the start-stop keyword.

**Examples** The debug output below shows the selection of a preferred server and the processing of requests for the configuration above.

```
Router#
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002C):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Server[0] load:0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Selected Server[0] with load 0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002C):Server (192.0.2.238:2095,2096) now being
 used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002D):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002D):Server (192.0.2.238:2095,2096) now being
 used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002E):No preferred server available.
```

```
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[3] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002E):Server (192.0.2.238:2095,2096) now being
 used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002F):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[2] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002F):Server (192.0.2.238:2095,2096) now being
 used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(00000030):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[1] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(00000030):Server (192.0.2.238:2095,2096) now being
 used as preferred server
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000031):No preferred server available.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new
server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Server[0] load:5
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Selected Server[1] with load 0
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000031):Server (192.0.2.238:2015,2016) now being
 used as preferred server
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000032):No preferred server available.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing
server.
.
.
.
```

Server Status Information for Named RADIUS Server Group Example

The output below shows the AAA server status for the named RADIUS server group configuration example.

```
Router# show aaa servers
RADIUS:id 8, priority 1, host 192.0.2.238, auth-port 2095, acct-port 2096
     State:current UP, duration 3781s, previous duration 0s
     Dead:total time 0s, count 0
     Quarantined:No
     Authen:request 0, timeouts 0
             Response:unexpected 0, server error 0, incorrect 0, time 0ms
             Transaction:success 0, failure 0
     Author:request 0, timeouts 0
             Response:unexpected 0, server error 0, incorrect 0, time 0ms
             Transaction:success 0, failure 0
     Account:request 0, timeouts 0
             Response:unexpected 0, server error 0, incorrect 0, time 0ms
             Transaction:success 0, failure 0
     Elapsed time since counters last cleared:0m
RADIUS:id 9, priority 2, host 192.0.2.238, auth-port 2015, acct-port 2016
     State:current UP, duration 3781s, previous duration 0s
     Dead:total time 0s, count 0
     Quarantined:No
     Authen:request 0, timeouts 0
             Response:unexpected 0, server error 0, incorrect 0, time 0ms
             Transaction:success 0, failure 0
     Author:request 0, timeouts 0
             Response:unexpected 0, server error 0, incorrect 0, time 0ms
             Transaction:success 0, failure 0
     Account:request 0, timeouts 0
             Response:unexpected 0, server error 0, incorrect 0, time 0ms
             Transaction:success 0, failure 0
     Elapsed time since counters last cleared:0m
Router#
```

The output shows the status of two RADIUS servers. Both servers are alive, and no requests have been processed since the counters were cleared 0 minutes ago.

## Related Commands

| Command | Description |
|---|---|
| **debug aaa sg-server selection** | Shows why the RADIUS and TACACS+ server group system in a router is selecting a particular server. |
| **debug aaa test** | Shows when the idle timer or dead timer has expired for RADIUS load balancing. |
| **radius-server host** | Enables RADIUS automated testing for load balancing. |
| **radius-server load-balance** | Enables RADIUS server load balancing for the global RADIUS server group. |
| **test aaa group** | Tests RADIUS load balancing server response manually. |