data

To configure the data interface type and number for a redundancy group, use the **data** command in redundancy application group configuration mode. To remove the configuration, use the **no** form of this command.

data *interface-type interface-number*

no data *interface-type interface-number*

Syntax Description

interface-type	Interface type.
interface-number	Interface number.

Command Default

No data interface is configured.

Command Modes

Redundancy application group configuration (config-red-app-grp)

Command History

Release	Modification
Cisco IOS XE Release 3.1S	This command was introduced.

Usage Guidelines

Use the **data** command to configure the data interface. The data interface can be the same physical interface as the control interface.

Examples

The following example shows how to configure the data Gigabit Ethernet interface for group1:

```
Router# configure terminal
Router(config)# redundancy
Router(config-red)# application redundancy
Router(config-red-app)# group 1
Router(config-red-app-grp)# data GigabitEthernet 0/0/0
```

Related Commands

Command	Description
application redundancy	Enters redundancy application configuration mode.
authentication	Configures clear text authentication and MD5 authentication for a redundancy group.
control	Configures the control interface type and number for a redundancy group.
group(firewall)	Enters redundancy application group configuration mode.
name	Configures the redundancy group with a name.

Command	Description
preempt	Enables preemption on the redundancy group.
protocol	Defines a protocol instance in a redundancy group.

database archive

To set the certification authority (CA) certificate and CA key archive format—and the password—to encrypt this CA certificate and CA key archive file, use the **database archive** command in certificate server configuration mode. To disable the autoarchive feature, use the **no** form of this command.

database archive {**pkcs12** | **pem**} [**password** *password*]

no database archive {**pkcs12** | **pem**} [**password** *password*]

Syntax Description

pkcs12	Export as a PKCS12 file. The default is PKCS12.
pem	Export as a privacy-enhanced mail (PEM) file.
password *password*	(Optional) Password to encrypt the CA certificate and CA key. The password must be at least eight characters. If a password is not specified, you will be prompted for the password after the **no shutdown** command has been issued for the first time. When the password is entered, it will be encrypted.

Defaults

The archive format is PKCS (that is, the CA certificate and CA key are exported into a PKCS12 file, and you will be prompted for the password when the certificate server is turned on the first time).

Command Modes

Certificate server configuration

Command History

Release	Modification
12.3(11)T	This command was introduced.

Usage Guidelines

Use this command to configure the autoarchive format for the CA certificate and CA key. The archive can later be used to restore your certificate server.

If autoarchiving is not explicitly turned off when the certificate server is first enabled (using the **no shutdown** command), the CA certificate and CA key will be archived automatically, applying the following rule:

- The CA key must be (1) manually generated and marked “exportable” or (2) automatically generated by the certificate server (it will be marked nonexportable).

Note It is strongly recommended that if the password is included in the configuration to suppress the prompt after the **no shutdown** command, the password should be removed from the configuration after the archiving is finished.

Examples

The following example shows that certificate server autoarchiving has been enabled. The CA certificate and CA key format has been set to PEM, and the password has been set as cisco123.

```
Router (config)# crypto pki server myserver
Router (cs-server)# database archive pem password cisco123
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server.

database level

To control what type of data is stored in the certificate enrollment database, use the **database level** command in certificate server configuration mode. To return to the default functionality, use the **no** form of this command.

database level {**minimal** | **names** | **complete**}

no database level {**minimal** | **names** | **complete**}

Syntax Description

minimal	Enough information is stored only to continue issuing new certificates without conflict. This is the default functionality.
names	The serial number and subject name of each certificate are stored in the database, providing enough information for the administrator to find and revoke and particular certificate, if necessary.
complete	Each issued certificate is written to the database. If this keyword is used, you should enable the **database url** command; see "Usage Guidelines" for more information.

Defaults

minimal

Command Modes

Certificate server configuration

Command History

Release	Modification
12.3(4)T	This command was introduced.

Usage Guidelines

The **database level** command is used to describe the database of certificates and certification authority (CA) states. After the user downgrades the database level, the old data stays the same and the new data is logged at the new level.

minimum Level

The *ca-label*.ser file is always available. It contains the previously issued certificate's serial number, which is always 1. If the .ser file is unavailable and the CA server has a self-signed certificate in the local configuration, the CA server will refuse to issue new certificates.

The file format is as follows:

```
last_serial = serial-number
```

names Level

The *serial-number*.cnm file, which is written for each issued certificate, contains the "human readable decoded subject name" of the issued certificate and the "der encoded" values. This file can also include a certificate expiration date and the current status. (The **minimum** level files are also written out.)

The file format is as follows:

```
subjectname_der = <base64 encoded der value>
subjectname_str = <human readable decode subjectname>
expiration = <expiration date>
status = valid | revoked
```

complete Level

The *serial-number*.cer file, which is written for each issued certificate, is the binary certificate without additional encoding. (The **minimum** and **names** level files are also written out.)

The **complete** level produces a large amount of information, so you may want to store all database entries on an external TFTP server via the **database url** command unless your router does one of the following:

- Issues only a small number of certificates
- Has a local file system that is designed to support a large number of write operations and has sufficient storage for the certificates that are being issued

Examples

The following example shows how configure a minimum database to be stored on the local system:

```
Router#(config) ip http server
Router#(config) crypto pki server myserver
Router#(cs-server) database level minimum
Router#(cs-server) database url nvram:
Router#(cs-server) issuer-name CN = ipsec_cs,L = Santa Cruz,C = US
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI configuration mode.
database url	Specifies the location where all database entries for the certificate server will be written out.

database url

To specify the location where database entries for the certificate server (CS) will be stored or published, use the **database url** command in certificate server configuration mode. To return to the default location, use the **no** form of this command.

Storing Files to a Primary Location

database url *root-url*

Storing Critical CS Files to a Specific Location

database url [{**cnm** | **crl** | **crt** | **p12** | **pem** | **ser**}] *root-url* [**username** *username*] [**password** [*encrypt-type*] *password*]

no database url [{**cnm** | **crl** | **crt** | **p12** | **pem** | **ser**}] *root-url* [**username** *username*] [**password** [*encrypt-type*] *password*]

Publishing Noncritical CS Files to a Specific Location

database url {**cnm** | **crl** | **crt**} **publish** *root-url* [**username** *username*][**password** [*encrypt-type*] *password*]

no database url {**cnm** | **crl** | **crt**} **publish** *root-url* [**username** *username*][**password** [*encrypt-type*] *password*]

Syntax Description

root-url	Location where database entries will be written out. The URL can be any URL that is supported by the Cisco IOS file system (IFS).
cnm	(Optional) Specifies the certificate name and expiration file to be stored or published to a specific location.
crl	(Optional) Specifies the DER-encoded certificate revocation list to be stored or published to a specific location
crt	(Optional) Specifies the DER-encoded certificate files to be stored or published to a specific location.
p12	(Optional) Specifies the CS certificate and key archive file in PKCS12 format to be stored to a specific location.
pem	(Optional) Specifies the CS certificate and key archive file in privacy-enhanced mail format to be stored to a specific location.
ser	(Optional) Specifies the current serial number to be stored to a specific location.
publish	Specifies that the files will be made available to a published location.
username *username*	(Optional) When prompted, a username will be used to access a storage location.

password *password*	(Optional) When prompted, a password will be used to access a storage location.
encrypt-type	(Optional) Type of encryption to be used for the password. If no password type is specified the password is sent as clear text. • Default is 0; specifies that the password entered will be encrypted. • 7; specifies that the password entered is already encrypted.

Defaults

The default file storage location is flash.

No default file publish location is specified.

Command Modes

Certificate server configuration (cs-server)

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.4(4)T	This command was modified. The following keywords and arguments were added **cnm**, **crl**, **crt**, **p12**, **pem**, **ser**, **publish, username** *username***,** *encrypt-type* and **password** *password*.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.(33)SRA.

Usage Guidelines

After you create a certificate server via the **crypto pki server** command, use the **database url** command if you want to specify a combined list of all the certificates that have been issued and the current command revocation list (CRL). The CRL is written to the certificate enrollment database as *ca-label*.crl (where *ca-label* is the name of the certificate server).

Note Although issuing the **database url** command is not required, it is recommended. Unless your router has a local file system that is designed for a large number of write operations and has sufficient storage for the certificates that are issued, you should issue this command.

Cisco IOS File System

The router uses any file system that is supported by your version of Cisco IOS software (such as TFTP, FTP, flash, and NVRAM) to send a certificate request and to receive the issued certificate. A user may wish to enable IFS certificate enrollment when his or her certification authority (CA) does not support Simple Certificate Enrollment Protocol (SCEP).

Specifying CS Storage and Publication Location by File Type

The CS allows the flexibility to store different critical file types to specific storage locations and publish non-critical files to the same or alternate locations. When choosing storage locations consider the file security needed and server performance. For instance, serial number files (.ser) and archive files (.p12 or .pem) might have greater security restrictions than the general certificates storage location (.crt) or the name file storage location (.cnm). Performance of your certificate server may be affected by the storage location(s) you choose, for example, reading from a network location would likely take more time than reading directly from a router's local storage device.

Examples

The following example shows how to configure all database entries to be written out to a TFTP server:

```
Router#(config) ip http server
Router#(config) crypto pki server myserver
Router#(cs-server) database level complete
Router#(cs-server) database url tftp://mytftp
```

The following example shows the configuration of a primary storage location for critical files, a specific storage location for the critical file serial number file, the main CS database file, and a password protected file publication location for the CRL file:

```
Router(config)# crypto pki server mycs
Router(cs-server)# database url ftp://cs-db.company.com
!
% Server database url was changed. You need to move the
% existing database to the new location.
!
Router(cs-server)# database url ser nvram:
Router(cs-server)# database url crl publish ftp://crl.company.com username myname password
mypassword
Router(cs-server)# end
```

The following show output displays the specified primary storage location and critical file storage locations specified:

```
Router# show

Sep  3 20:19:34.216: %SYS-5-CONFIG_I: Configured from console by user on console
Router# show crypto pki server

Certificate Server mycs:
     Status: disabled
     Server's configuration is unlocked  (enter "no shut" to lock it)
     Issuer name: CN=mycs
     CA cert fingerprint: -Not found-
     Granting mode is: manual
     Last certificate issued serial number: 0x0
     CA certificate expiration timer: 00:00:00 GMT Jan 1 1970
     CRL not present.
     Current primary storage dir: ftp://cs-db.company.com
     Current storage dir for .ser files: nvram:
     Database Level: Minimum - no cert data written to storage
Router#
```

The following show output displays all storage and publication locations. The serial number file (.ser) is stored in NVRAM. The CRL file will be published to ftp://crl.company.com with a username and password. All other critical files will be stored to the primary location, ftp://cs-db.company.com.

```
Router# show running-config

   section crypto pki server
   crypto pki server mycs shutdown database url ftp://cs-db.company.com
   database url crl publish ftp://crl.company.com username myname password 7
12141C0713181F13253920
   database url ser nvram:
Router#
```

Verifying the Database URL

To ensure that the specified URL is working correctly, configure the **database url** command before you issue the **no shutdown** command on the certificate server for the first time. If the URL is broken, you will see output as follows:

```
Router(config)# crypto pki server mycs
Router(cs-server)# database url ftp://myftpserver
Router(cs-server)# no shutdown
% Once you start the server, you can no longer change some of
% the configuration.
Are you sure you want to do this? [yes/no]: yes
Translating "myftpserver"
% There was a problem reading the file 'mycs.ser' from certificate storage.
% Please verify storage accessibility and enable the server again.

% Failed to generate CA certificate - 0xFFFFFFFF
% The Certificate Server has been disabled.
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI server configuration mode.
database level	Controls what type of data is stored in the database.
database username	Requires a username or password to be issued when accessing the primary database storage location.

database username

To require a username or password to be issued when accessing the primary database location, use the **database username** command in certificate server configuration mode. To return to the default value, use the **no** form of this command.

database username *username* [**password** [*encr-type*] *password*]

no database username *username* [**password** [*encr-type*] *password*]

Syntax Description

username	When prompted, a username will be used to access a storage location.
password *password*	(Optional) When prompted, a password will be used to access a storage location.
encr-type	(Optional) Type of encryption to be used for the password. If no password encryption type is specified, the password is sent as clear text. • Default is 0; specifies that the password entered will be encrypted. • 7; specifies the password entered is already encrypted.

Defaults

No username or password will be used to access the primary database storage location.

Command Modes

Certificate server configuration

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.4(4)T	The command name was changed from **database** (certificate server) to **database username**.

Usage Guidelines

All information stored in the remote database is public: there are no private keys stored in the database location. Using a password helps to protect against a potential attacker who can change the contents of the .ser or .crl file. If the contents of the files are changed, the certificate server may shut down, refusing to either issue new certificates or respond to Simple Certificate Enrollment Protocol (SCEP) requests until the files are restored.

It is good security practice to protect all information exchanges with the database server using IP Security (IPsec). To protect your information, use a remote database to obtain the appropriate certificates and setup the necessary IPsec connections to protect all future access to the database server.

Examples

The following example shows how to specify the username “mystorage” when the primary storage location is on an external TFTP server:

```
Router (config)# ip http server
Router (config)# crypto pki server myserver
Router (cs-server)# database level complete
```

```
Router (cs-server)# database url tftp://mytftp
Router (cs-server)# database username mystorage
```

Related Commands

Command	Description
crypto pki server	Enables a Cisco IOS certificate server and enters PKI server configuration mode.
database level	Controls what type of data is stored in the database.
database url	Specifies the primary storage location for the certificate server.

deadtime (server-group configuration)

To configure deadtime within the context of RADIUS server groups, use the **deadtime** command in server group configuration mode. To set deadtime to 0, use the **no** form of this command.

deadtime *minutes*

no deadtime

Syntax Description

minutes	Length of time, in minutes, for which a RADIUS server is skipped over by transaction requests, up to a maximum of 1440 minutes (24 hours).

Defaults

Deadtime is set to 0.

Command Modes

Server-group configuration

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Use this command to configure the deadtime value of any RADIUS server group. The value of deadtime set in the server groups will override the server that is configured globally. If deadtime is omitted from the server group configuration, the value will be inherited from the master list. If the server group is not configured, the default value (0) will apply to all servers in the group.

When the RADIUS Server Is Marked As Dead

For Cisco IOS versions prior to 12.2(13.7)T, the RADIUS server will be marked as dead if a transaction is transmitted for the configured number of retransmits and a valid response is not received from the server within the configured timeout for any of the RADIUS packet transmissions.

For Cisco IOS versions 12.2(13.7)T and later, the RADIUS server will be marked as dead if both of the following conditions are met:

1. A valid response has not been received from the RADIUS server for any outstanding transaction for at least the timeout period that is used to determine whether to retransmit to that server, and
2. Across all transactions being sent to the RADIUS server, at least the requisite number of retransmits +1 (for the initial transmission) have been sent consecutively without receiving a valid response from the server with the requisite timeout.

Examples

The following example specifies a one-minute deadtime for RADIUS server group group1 once it has failed to respond to authentication requests:

```
aaa group server radius group1
 server 10.1.1.1 auth-port 1645 acct-port 1646
 server 10.2.2.2 auth-port 2000 acct-port 2001
 deadtime 1
```

Related Commands

Command	Description
radius-server deadtime	Sets the deadtime value globally.

default (ca-trustpoint)

To reset the value of a ca-trustpoint configuration subcommand to its default, use the **default** command in ca-trustpoint configuration mode.

default *command-name*

Syntax Description

command-name	Ca-trustpoint configuration subcommand.

Defaults

No default behavior or values.

Command Modes

Ca-trustpoint configuration

Command History

Release	Modification
12.1(1)T	This command was introduced.
12.2(8)T	The command mode was changed from **default (ca-root)** to **default (ca-trustpoint)** to support the **crypto ca trustpoint** command and all related subcommands.
12.2(18)SXD	The **default (ca-root)** command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(33)SRA	The **default (ca-root)** command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Before you can configure this command, you must enable the **crypto ca trustpoint** command, which enters ca-trustpoint configuration mode.

Use this command to reset the value of a ca-trustpoint configuration mode subcommand to its default.

Note The **crypto ca trustpoint** command deprecates the **crypto ca identity** and **crypto ca trusted-root** commands and all related subcommands (all ca-identity and trusted-root configuration mode commands). If you enter a ca-identity or trusted-root subcommand, the configuration mode and command will be written back as ca-trustpoint.

Examples

The following example shows how to remove the **crl optional** command from your configuration; the default of **crl optional** is off.

```
default crl optional
```

Related Commands

Command	Description
crypto ca trustpoint	Declares the CA that your router should use.

default-group-policy

To associate a policy group with a SSL VPN context configuration, use the **default-group-policy** command in webvpn context configuration mode. To remove the policy group from the webvpn context configuration, use the **no** form of this command.

default-group-policy *name*

no default-group-policy

Syntax Description

name	Name of the policy configured with the **policy group** command.

Command Default

A policy group is not associated with a SSL VPN context configuration.

Command Modes

Webvpn context configuration

Command History

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

Usage Guidelines

The **policy group** command is first configured to define policy group configuration parameters. This command is configured to attach the policy group to the SSL VPN context when multiple policy groups are defined under the context. This policy will be used as the default unless an authentication, authorization, and accounting (AAA) server pushes an attribute that specifically requests another group policy.

Examples

The following example configures policy group ONE as the default policy group:

```
Router(config)# webvpn context context1
Router(config-webvpn-context)# policy-group ONE
Router(config-webvpn-group)# exit
Router(config-webvpn-context)# policy-group TWO
Router(config-webvpn-group)# exit
Router(config-webvpn-context)# default-group-policy ONE
```

Related Commands

Command	Description
policy group	Enters webvpn group policy configuration mode to configure a policy group.
webvpn context	Enters webvpn context configuration mode to configure the SSL VPN context.

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* {{*source-addr source-wildcard*} | **object-group** *object-group-name* | **any** | **host** {*address* | *name*}} {*destination-addr destination-wildcard*} | **object-group** *object-group-name* | **any** | **host** {*address* | *name*}}

deny {**tcp** | **udp**} {{*source-addr source-wildcard*} | **object-group** *source-addr-group-name* | **any** | **host** {*address* | *name*} {*destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *srcport-groupname*} {**object-group** *dest-addr-groupname* | *destination* | *destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *destport-groupname*} [**dscp** *type*] [**fragments**] [**option** *option*] [**precedence** *precedence*] [**log**] [**log-input**] [**time-range** *time-range-name*] [**tos** *tos*]]}

no deny *protocol* {{*source-addr source-wildcard*} | **object-group** *object-group-name* | **any** | **host** {*address* | *name*}} {*destination-addr destination-wildcard*} | **object-group** *object-group-name* | **any** | **host** {*address* | *name*}}

no deny {**tcp** | **udp**} {{*source-addr source-wildcard*} | **object-group** *source-addr-group-name* | **any** | **host** {*address* | *name*} {*destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *srcport-groupname*} {**object-group** *dest-addr-groupname* | *destination* | *destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *destport-groupname*} [**dscp** *type*] [**fragments**] [**option** *option*] [**precedence** *precedence*] [**log**] [**log-input**] [**time-range** *time-range-name*] [**tos** *tos*]]}

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.
source-addr	Number of the network or host from which the packet is being sent in a 32-bit quantity in four-part, dotted-decimal format.
source-wildcard	Wildcard bits to be applied to source 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 *address*	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 "Fragments and Policy Routing" 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.) 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. 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. 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. 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 "Access List or OGACL Processing of Fragments" and "Fragments and Policy Routing" sections in the "Usage Guidelines" section.

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 Table 29:

Table 29 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: • The entry is applied to nonfragmented packets, initial fragments, and noninitial fragments. For an access list entry containing Layer 3 and Layer 4 information: • The entry is applied to nonfragmented packets and initial fragments: – If the entry is a **permit** statement, the packet or fragment is permitted. – If the entry is a **deny** statement, the packet or fragment is denied. • 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 – If the entry is a **permit** statement, the noninitial fragment is permitted. – If the entry is a **deny** statement, the next access-list entry is processed. **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** 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.
show object-group	Displays information about object groups that are configured.

deny (Catalyst 6500 series switches)

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

deny *protocol* {{*source-addr source-wildcard*} | **addrgroup** *object-group-name* | **any** | **host** {*address* | *name*}} {*destination-addr destination-wildcard*} | **addrgroup** *object-group-name* | **any** | **host** {*address* | *name*}}

deny {**tcp** | **udp**} {{*source-addr source-wildcard*} | **addrgroup** *source-addr-group-name* | **any** | **host** {*address* | *name*} {*destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *srcport-groupname*} {**addrgroup** *dest-addr-groupname* | *destination* | *destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *destport-groupname*} [**dscp** *type*] [**fragments**] [**option** *option*] [**precedence** *precedence*] [**log**] [**log-input**] [**time-range** *time-range-name*] [**tos** *tos*]]}

no deny *protocol* {{*source-addr source-wildcard*} | **addrgroup** *object-group-name* | **any** | **host** {*address* | *name*}} {*destination-addr destination-wildcard*} | **addrgroup** *object-group-name* | **any** | **host** {*address* | *name*}}

no deny {**tcp** | **udp**} {{*source-addr source-wildcard*} | **addrgroup** *source-addr-group-name* | **any** | **host** {*address* | *name*} {*destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *srcport-groupname*} {**addrgroup** *dest-addr-groupname* | *destination* | *destination-addr destination-wildcard* | **any** | **eq** *port* | **gt** *port* | **host** {*address* | *name*} | **lt** *port* | **neq** *port* | **portgroup** *destport-groupname*} [**dscp** *type*] [**fragments**] [**option** *option*] [**precedence** *precedence*] [**log**] [**log-input**] [**time-range** *time-range-name*] [**tos** *tos*]]}

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.
source-addr	Number of the network or host from which the packet is being sent in a 32-bit quantity in four-part, dotted-decimal format.
source-wildcard	Wildcard bits to be applied to source in four-part, dotted-decimal format. Place ones in the bit positions you want to ignore.
addrgroup *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 *address*	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.

addrgroup *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.
addrgroup *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 "Fragments and Policy Routing" 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" sectionfor 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.) 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. 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. 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. 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 due to 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 "Access List Processing of Fragments" and "Fragments and Policy Routing" sections in the "Usage Guidelines" section.

Command Default

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

Command Modes

Access-list configuration (config-ext-nacl)

Command History

Release	Modification
12.2(33)SXH	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 named 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 **addrgroup** *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 Protoc (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 Processing of Fragments

The behavior of access-list entries regarding the use or lack of the **fragments** keyword are summarized in Table 29:

Table 30 Access list 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: • The entry is applied to nonfragmented packets, initial fragments, and noninitial fragments. For an access list entry containing Layer 3 and Layer 4 information: • The entry is applied to nonfragmented packets and initial fragments: – If the entry is a **permit** statement, the packet or fragment is permitted. – If the entry is a **deny** statement, the packet or fragment is denied. • 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 – If the entry is a **permit** statement, the noninitial fragment is permitted. – If the entry is a **deny** statement, the next access-list entry is processed. **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(config)# ip access-list extended my-pbacl-policy
Router(config-ext-nacl)# deny tcp any any
Router(config-ext-nacl)# exit
Router(config)# exit
```

Related Commands

Command	Description
ip access-group	Controls access to an interface.
ip access-list	Defines an IP access list by name.
logging console	Limits messages logged to the console based on severity.
object-group	Defines object groups to optimize your configuration
permit (Catalyst 6500 series switches)	Sets conditions for a named IP access list.
show ip access-lists	Displays the contents of all current IP access lists.

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: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the **any** keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. • 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: • Use a 32-bit quantity in four-part dotted-decimal format. Place 1s in the bit positions that you want to ignore. • Use the **any** keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. • 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. **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: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the **any** keyword as an abbreviation for the *destination* and *destination-wildcard* of 0.0.0.0 255.255.255.255. • 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: • Use a 32-bit quantity in four-part dotted-decimal format. Place 1s in the bit positions that you want to ignore. • Use the **any** keyword as an abbreviation for a *destination* and *destination-wildcard* of 0.0.0.0 255.255.255.255. • 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 Table 31 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. • The *operator* can be **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), or **range** (inclusive range). • The *value* can range from 0 to 255. • If the operator is **range**, specify two values separated by a space. • For Release 12.0S, if the operator is **eq** or **neq**, only one TTL value can be specified. • 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 "Access List Processing of Fragments" and "Fragments and Policy Routing" 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). 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. 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. 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. **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**.

Defaults

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.
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 Table 31.

Table 31 IP Option Values and Names

IP Option Value or Name	Description
0 to 255	IP Options values.
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).
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: • The entry is applied to nonfragmented packets, initial fragments, and noninitial fragments. For an access list entry that contains Layer 3 and Layer 4 information: • The entry is applied to nonfragmented packets and initial fragments. – If the entry is a **permit** statement, then the packet or fragment is permitted. – If the entry is a **deny** statement, then the packet or fragment is denied. • 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 – If the entry is a **permit** statement, then the noninitial fragment is permitted. – If the entry is a **deny** statement, then the next access list entry is processed. **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.

Command	Description
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.
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 (MAC ACL)

To set conditions for a MAC access list, use the **deny** command in MAC access-list extended configuration mode. To remove a condition from an access list, use the **no** form of this command.

deny {*src_mac_mask* | {**host name** *src_mac_name*} | **any**} {*dest_mac_mask* | {**host name** *dst_mac_name*} | **any**} [{*protocol_keyword* | {*ethertype_number ethertype_mask*}} [**vlan** *vlan_ID*] [**cos** *cos_value*]]

no deny {*src_mac_mask* | {**host name** *src_mac_name*} | **any**} {*dest_mac_mask* | {**host name** *dst_mac_name*} | **any**} [{*protocol_keyword* | {*ethertype_number ethertype_mask*}} [**vlan** *vlan_ID*] [**cos** *cos_value*]]

Syntax Description

src_mac_mask	Specifies the MAC address mask that identifies a selected block of source MAC addresses. A value of 1 represents a wildcard in that position.
host name *src_mac_name*	Specifies a source host that has been named using the **mac host name** command.
any	Specifies any source or any destination host as an abbreviation for the *src_mac_mask* or *dest_mac_mask* value of 1111.1111.1111, which declares all digits to be wildcards.
dest_mac_mask	Specifies the MAC address mask that identifies a selected block of destination MAC addresses.
host name *dst_mac_name*	Specifies a destination host that has been named using the **mac host name** command.
protocol_keyword	(Optional) Specifies a named protocol (for example, ARP).
ethertype_number	(Optional) The EtherType number specifies the protocol within the Ethernet packet.
ethertype_mask	(Optional) The EtherType mask allows a range of EtherTypes to be specified together. This is a hexadecimal number from 0 to FFFF. An EtherType mask of 0 requires an exact match of the EtherType.
vlan *vlan_ID*	(Optional) Specifies a VLAN.
cos *cos_value*	(Optional) Specifies the Layer 2 priority level for packets. The range is from 0 to 7.

Command Default

This command has no defaults.

Command Modes

MAC access-list extended configuration (config-ext-macl)

Command History

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

Usage Guidelines

Use this command following the **ip access-list** command to define the conditions under which a packet passes the access list.

- The **vlan** and **cos** keywords are not supported in MAC ACLs used for VACL filtering.
- The **vlan** keyword for VLAN-based QoS filtering in MAC ACLs can be globally enabled or disabled and is disabled by default.
- Enter MAC addresses as three 2-byte values in dotted hexadecimal format. For example, 0123.4567.89ab.
- Enter MAC address masks as three 2-byte values in dotted hexadecimal format. Use 1 bits as wildcards. For example, to match an address exactly, use 0000.0000.0000 (can be entered as 0.0.0).
- An entry without a protocol parameter matches any protocol.
- Enter an EtherType and an EtherType mask as hexadecimal values from 0 to FFFF.
- This list shows the EtherType values and their corresponding protocol keywords:
  - 0x0600—xns-idp—Xerox XNS IDP
  - 0x0BAD—vines-ip—Banyan VINES IP
  - 0x0baf—vines-echo—Banyan VINES Echo
  - 0x6000—etype-6000—DEC unassigned, experimental
  - 0x6001—mop-dump—DEC Maintenance Operation Protocol (MOP) Dump/Load Assistance
  - 0x6002—mop-console—DEC MOP Remote Console
  - 0x6003—decnet-iv—DEC DECnet Phase IV Route
  - 0x6004—lat—DEC Local Area Transport (LAT)
  - 0x6005—diagnostic—DEC DECnet Diagnostics
  - 0x6007—lavc-sca—DEC Local-Area VAX Cluster (LAVC), SCA
  - 0x6008—amber—DEC AMBER
  - 0x6009—mumps—DEC MUMPS
  - 0x0800—ip—Malformed, invalid, or deliberately corrupt IP frames
  - 0x8038—dec-spanning—DEC LANBridge Management
  - 0x8039—dsm—DEC DSM/DDP
  - 0x8040—netbios—DEC PATHWORKS DECnet NETBIOS Emulation
  - 0x8041—msdos—DEC Local Area System Transport
  - 0x8042—etype-8042—DEC unassigned
  - 0x809B—appletalk—Kinetics EtherTalk (AppleTalk over Ethernet)
  - 0x80F3—aarp—Kinetics AppleTalk Address Resolution Protocol (AARP)

Examples

This example shows how to create a MAC-Layer ACL named mac_layer that denies dec-phase-iv traffic with source address 0000.4700.0001 and destination address 0000.4700.0009, but allows all other traffic:

```
Router(config)# mac access-list extended mac_layer
Router(config-ext-macl)# deny 0000.4700.0001 0.0.0 0000.4700.0009 0.0.0 dec-phase-iv
Router(config-ext-macl)# permit any any
```

Related Commands

Command	Description
permit (MAC ACL)	Sets permit conditions for a named MAC access list.
mac access-list extended	Defines a MAC access list by name.
mac host	Assigns a name to a MAC address.
show mac access-group	Displays the contents of all current MAC access groups.

deny (WebVPN)

To set conditions in a named Secure Sockets Layer Virtual Private Network (SSL VPN) access list that will deny packets, use the **deny** command in webvpn acl configuration mode. To remove a deny condition from an access list, use the **no** form of this command.

deny [**url** [**any** | *url-string*]] [**ip** | **tcp** | **udp** | **http** | **https** | **cifs**] [**any** | *source-ip source-mask*] [**any** | *destination-ip destination-mask*] [**time-range** *time-range-name*] [**syslog**]

no deny url [**any** | *url-string*] [**ip** | **tcp** | **udp** | **http** | **https** | **cifs**] [**any** | *source-ip source-mask*] [**any** | *destination-ip destination-mask*] [**time-range** *time-range-name*] [**syslog**]

Syntax Description

url	(Optional) Filtering rules are applied to the URL. • Use the **any** keyword as an abbreviation for any URL.
url-string	(Optional) URL string defined as follows: **scheme://host[:port][/path]** • **scheme**—Can be HTTP, Secure HTTPS (HTTPS), or Common Internet File System (CIFS). This field is required in the URL string. • **host**—Can be a hostname or a host IP (host mask). The host can have one wildcard (*). • **port**—Can be any valid port number (1–65535). It is possible to have multiple port numbers separated by a comma (,). The port range is expressed using a dash (-). • **path**—Can be any valid path string. In the path string, the $user is translated to the current user name.
ip	(Optional) Denies only IP packets. When you enter the **ip** keyword, you must use the specific command syntax shown for the IP form of the deny command.
tcp	(Optional) 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	(Optional) 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.
http	(Optional) Denies only HTTP packets. When you enter the **http** keyword, you must use the specific command syntax shown for the HTTP form of the deny command.
https	(Optional) Denies only HTTPS packets. When you enter the **https** keyword, you must use the specific command syntax shown for the HTTPS form of the deny command.
cifs	(Optional) Denies only CIFS packets. When you enter the **cifs** keyword, you must use the specific command syntax shown for the CIFS form of the deny command.
source-ip source-mask	(Optional) Number of the network or host from which the packet is being sent. There are three alternative ways to specify the source: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the **any** keyword as an abbreviation for a source and source mask of 0.0.0.0 255.255.255.255. • Use **host** source as an abbreviation for a source and source-wildcard of source 0.0.0.0.

destination-ip *destination-mask*	(Optional) Number of the network or host to which the packet is being sent. There are three alternative ways to specify the destination: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the **any** keyword as an abbreviation for a source and source mask of 0.0.0.0 255.255.255.255. • Use **host** source as an abbreviation for a source and source-wildcard of source 0.0.0.0.
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.
syslog	(Optional) System logging messages are generated.

Command Default

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

Command Modes

Webvpn acl configuration

Command History

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

Usage Guidelines

Use this command following the **acl** 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.

Examples

The following example shows that all packets from the URL "https://10.168.2.228:34,80-90,100-/public" will be denied:

```
webvpn context context1
 acl acl1
  deny url "https://10.168.2.228:34,80-90,100-/public"
```

Related Commands

Command	Description
absolute	Specifies an absolute time for a time range.
periodic	Specifies a recurring (weekly) time range for functions that support the time-range feature.
permit (webvpn acl)	Sets conditions to allow a packet to pass a named SSL VPN access list.
time-range	Enables time-range configuration mode and defines time ranges for functions (such as extended access lists).

description (dot1x credentials)

To specify a description for an 802.1X profile, use the **description** command in dot1x credentials configuration mode. To remove the description, use the **no** form of this command.

description *text*

no description

Syntax Description

text	Text description. The description can be up to 80 characters.

Command Default

A description is not specified.

Command Modes

Dot1x credentials configuration

Command History

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

Usage Guidelines

Before using this command, the **dot1x credentials** command must have been configured.

An 802.1X credential structure is necessary when configuring a supplicant (client). 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, and it provides a description of the credentials profile:

```
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
dot1x credentials	Specifies which 802.1X credentials profile to use.

description (identify zone)

To enter a description of a zone, use the **description** command in security zone configuration mode. To remove the description of the zone, use the **no** form of this command.

description *line-of-description*

no description *line-of-description*

Syntax Description

line-of-description	Description of the zone. You can enter up to 40 characters.

Command Default

None

Command Modes

Security zone configuration

Command History

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

Usage Guidelines

You can use this subcommand after entering the **zone security** or **zone-pair security** command.

Examples

The following example specifies that zone z1 is a testzone:

```
zone security z1
 description testzone
```

Related Commands

Command	Description
zone-pair security	Creates a zone-pair that is the type security.
zone security	Creates a zone.

description (identity policy)

To enter a description for an identity policy, use the **description** command in identity policy configuration mode. To remove the description, use the **no** form of this command.

description *line-of-description*

no description *line-of-description*

Syntax Description

line-of-description	Description of the identity policy.

Defaults

A description is not entered for the identity policy.

Command Modes

Identity policy configuration (config-identity-policy)

Command History

Release	Modification
12.3(8)T	This command was introduced.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.

Examples

The following example shows that a default identity policy and its description ("policyname1") have been specified:

```
Router (config)# identity policy policyname1
Router (config-identity-policy)# description policyABC
```

Related Commands

Command	Description
description (identity profile)	Enters a description for an identity profile.

description (identity profile)

To enter a description for an identity profile, use the **description** command in identity profile configuration mode. To remove the description of the identity profile, use the **no** form of this command.

description *line-of-description*

no description *line-of-description*

Syntax Description

line-of-description	Description of the identity profile.

Defaults

A description is not entered for the identity profile.

Command Modes

Identity profile configuration (config-identity-prof)

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	This command was previously configured in dot1x configuration mode.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.

Usage Guidelines

The **identity profile** command and one of its keywords (**default**, **dot1x**, or **eapoudp**) must be entered in global configuration mode before the **description** command can be used.

Examples

The following example shows that a default identity profile and its description (“ourdefaultpolicy”) have been specified:

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

Related Commands

Command	Description
description (identity policy)	Enters a description for an identity policy.
identity profile	Creates an identity profile and enters identity profile configuration mode.

description (IKEv2 keyring)

To add the description of an Internet Key Exchange Version 2 (IKEv2) peer or profile, use the **description** command in the IKEv2 keyring peer configuration mode. To delete the description, use the **no** form of this command.

description *line-of-description*

no description *line-of-description*

Syntax Description

line-of-description	Description given to an IKE peer or profile.

Command Default

The peer or profile is not described.

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.

Usage Guidelines

Use this command to provide a descriptive line about the IKEv2 peer, peer group, or profile.

Examples

The following example shows that the description "connection from site A" has been added to an IKEv2 peer:

```
Router(config)# crypto ikev2 keyring keyr 1
Router(configikev2-keyring)# peer peer1
Router(config-ikev2-keyring-peer)# description connection from site A
```

Related Commands

Command	Description
address (ikev2 keyring)	Specifies the IPv4 address or the range of the peers in IKEv2 keyring.
crypto ikev2 keyring	Defines an IKEv2 keyring.
hostname (ikev2 keyring)	Specifies the hostname for the peer in the IKEv2 keyring.
identity (ikev2 keyring)	Identifies the peer with IKEv2 types of identity.

Command	Description
peer	Defines a peer or a peer group for the keyring.
pre-shared-key (ikev2 keyring)	Defines a preshared key for the IKEv2 peer.

description (isakmp peer)

To add the description of an Internet Key Exchange (IKE) peer, use the **description** command in ISAKMP peer configuration mode. To delete the description, use the **no** form of this command.

description *line-of-description*

no description *line-of-description*

Syntax Description

line-of-description	Description given to an IKE peer.

Defaults

No default behavior or values

Command Modes

ISAKMP peer configuration

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.

Usage Guidelines

IKE peers that "sit" behind a Network Address Translation (NAT) device cannot be uniquely identified; therefore, they have to share the same peer description.

Examples

The following example shows that the description "connection from site A" has been added for an IKE peer:

```
Router# crypto isakmp peer address 10.2.2.9
Router (config-isakmp-peer)# description connection from site A
```

Related Commands

Command	Description
clear crypto session	Deletes crypto sessions (IPSec and IKE SAs).
show crypto isakmp peer	Displays peer descriptions.
show crypto session	Displays status information for active crypto sessions in a router.

destination host

To configure the fully qualified domain name (FQDN) of a Diameter peer, use the **destination host** command in diameter peer configuration submode. To disable the configured FQDN, use the **no** form of this command.

destination host *string*

no destination host *string*

Syntax Description

string	The FQDN of the Diameter peer.

Command Default

No FQDN is configured.

Command Modes

Diameter peer configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Examples

The following example shows how to configure the destination host:

```
Router(config-dia-peer)# destination host host1.example.com.
```

Related Commands

Command	Description
destination realm	Configures the destination realm of a Diameter peer.
diameter peer	Configures a Diameter peer and enters Diameter peer configuration submode.

destination realm

To configure the destination realm of a Diameter peer, use the **destination realm** command in diameter peer configuration submode. To disable the configured realm, use the **no** form of this command.

destination realm *string*

no destination realm *string*

Syntax Description

string	The destination realm (part of the domain @*realm*) in which a Diameter peer is located.

Command Default

No realm is configured.

Command Modes

Diameter peer configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

The realm might be added by the authentication, authorization, and accounting (AAA) client when sending a request to AAA. However, if the client does not add the attribute, then the value configured while in Diameter peer configuration submode is used when sending messages to the destination Diameter peer. If a value is not configured while in Diameter peer configuration submode, the value specified by the **diameter destination realm** global configuration command is used.

Examples

The following example shows how to configure the destination realm:

```
router (config-dia-peer)# destination realm example.com
```

Related Commands

Command	Description
diameter destination realm	Configures a global Diameter destination realm.
diameter peer	Configures a Diameter peer and enters Diameter peer configuration submode.

device (identity profile)

To statically authorize or reject individual devices, use the **device** command in identity profile configuration mode. To disable the authorization or rejection, use the **no** form of this command.

device {**authorize** {**ip address** *ip-address* **policy** *policy-name* | **mac-address** *mac-address* | **type** {**cisco** | **ip** | **phone**}} | **not-authorize**}

no device {**authorize** {**ip address** *ip-address* **policy** *policy-name* | **mac-address** *mac-address* | **type** {**cisco** | **ip** | **phone**}} | **not-authorize**}

Syntax Description

authorize	Configures an authorized device.
ip address	Specifies a device by its IP address.
ip-address	The IP address.
policy	Applies an associated policy with the device.
policy-name	Name of the policy.
mac-address	Specifies a device by its MAC address.
mac-address	The MAC address.
type	Specifies a device by its type.
cisco	Specifies a Cisco device.
ip	Specifies an IP device.
phone	Specifies a Cisco IP phone.
not-authorize	Configures an unauthorized device.

Defaults

A device is not statically authorized or rejected.

Command Modes

Identity profile configuration (config-identity-prof)

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 **unauthorize** keyword was changed to **not authorize**. The *cisco-device* argument was deleted. The **ip address** keyword and *ip-address* argument were added. The **ip** and **phone** keywords were added.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.

Usage Guidelines

The **identity profile** command and **default**, **dot1x**, or **eapoudp** keywords must be entered in global configuration mode before the **device** command can be used.

Examples

The following configuration example defines an identity profile for Extensible Authentication Protocol over User Datagram Protocol (EAPoUDP) to statically authorize host 192.168.1.3 with "policyname1" as the associated identity policy:

```
Router(config)# identity profile eapoudp
Router(config-identity-prof)# device authorize ip-address 192.168.1.3 policy policyname1
```

Related Commands

Command	Description
identity profile eapoudp	Creates an identity profile.

dhcp (IKEv2)

To assign an IP address to the remote access client using a DHCP server, use the **dhcp** command in IKEv2 authorization policy configuration mode. To remove the assigned IP address, use the **no** form of this command.

dhcp {**giaddr** *ip-address* | **server** {*ip-address* | *hostname*} | **timeout** *seconds*}

no dhcp {**giaddr** | **server** | **timeout**}

Syntax Description

giaddr *ip-address*	Specifies the gateway IP address (giaddr).
server	Specifies addresses for the DHCP server.
ip-address	IP address of the DHCP server.
hostname	Hostname of the DHCP server. The hostname is resolved during configuration.
timeout *seconds*	Specifies the wait time in seconds before the next DHCP server in the list is tried.

Command Default

An IP address is not assigned by a DHCP server.

Command Modes

IKEv2 client group configuration (config-ikev2-author-policy)

Command History

Release	Modification
15.1(3)T	This command was introduced.
Cisco IOS XE Release 3.3S	This command was integrated into Cisco IOS XE Release 3.3S.

Usage Guidelines

If this command is not configured, an IP address is assigned to a remote device using either a local pool that is configured on a router or a framed IP address attribute that is defined in RADIUS.

Note You can specify only one DHCP server.

Examples

The following example shows that the IP address of the DHCP server is 192.0.2.1 and that the time to wait until the next DHCP server on the list is tried is 6 seconds:

```
Router(config)# crypto ikev2 authorization policy home
Router(config-ikev2-client-config-group)# key abcd
Router(config-ikev2-client-config-group)# dhcp server 192.0.2.1
Router(config-ikev2-client-config-group)# dhcp timeout 6
```

Related Commands

Command	Description
crypto ikev2 authorization policy	Specifies an IKEv2 authorization policy group.

dhcp server (isakmp)

To assign an IP address or hostname using a DHCP server, use the **dhcp server** command in crypto ISAKMP group configuration mode. To remove the assigned IP address or hostname, use the **no** form of this command.

dhcp server {*ip-address* | *hostname*}

no dhcp server {*ip-address* | *hostname*}

Syntax Description

ip-address	Address of the DHCP server.
hostname	Hostname of the DHCP server.

Command Default

IP address is not assigned by a DHCP server.

Command Modes

Crypto ISAKMP group configuration (config-isakmp-group)

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

If this command is not configured, an IP address is assigned to a remote device using either a local pool that is configured on a router or a framed IP address attribute that is defined in RADIUS.

Note Up to five DHCP servers can be configured one at a time.

Note The DHCP proxy feature does not include functionality for the DHCP server to "push" the DNS, WINS server, or domain name to the remote client.

Examples

The following example shows that the IP address of the DHCP server is 10.2.3.4 and that the time to wait until the next DHCP server on the list is tried is 6 seconds:

```
Router (config)# crypto isakmp client configuration group home
 Router (config-isakmp-group)# key abcd
 Router (config-isakmp-group)# dhcp server 10.2.3.4
 Router (config-isakmp-group)# dhcp timeout 6
```

Related Commands

Command	Description
crypto isakmp client configuration group	Specifies to which group a policy profile will be defined.

dhcp timeout

To set the wait time before the next DHCP server on the list is tried, use the **dhcp timeout** command in crypto ISAKMP group configuration mode. To remove the wait time that was set, use the **no** form of this command.

dhcp timeout *time*

no dhcp timeout *time*

Syntax Description

time	Response time in seconds. Value = 4 through 30.

Command Modes

Crypto ISAKMP group configuration (config-isakmp-group)

Command History

Release	Modification
12.4(9)T	This command was introduced.

Examples

The following example shows that the IP address of the DHCP server is 10.2.3.4 and that the time to wait until the next DHCP server on the list is tried is 6 seconds:

```
Router (config)# crypto isakmp client configuration group home
 Router (config-isakmp-group)# dhcp server 10.2.3.4
 Router (config-isakmp-group)# key abcd
 Router (config-isakmp-group)# dhcp timeout 6
```

Related Commands

Command	Description
crypto isakmp client configuration group	Specifies to which group a policy profile will be defined.

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.

Defaults

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.

diameter origin host

To configure the fully qualified domain name (FQDN) of the host of a Diameter node, use the **diameter origin host** command in global configuration mode. To disable the configured FQDN, use the **no** form of this command.

diameter origin host *string*

no diameter origin host *string*

Syntax Description

string	Character string that describes the FQDN for a specific Diameter node.

Command Default

No realm is configured.

Command Modes

Global configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

Because there is no host configured by default, it is mandatory to configure this information. The origin host information is sent in requests to a Diameter peer. Global Diameter protocol parameters are used if Diameter parameters have not been defined at a Diameter peer level.

Examples

The following example shows how to configure a Diameter origin host:

```
Router(config)# diameter origin host host1.example.com.
```

Related Commands

Command	Description
diameter origin realm	Configures origin realm information for a Diameter node.
diameter peer	Defines a Diameter peer and enters Diameter peer configuration mode.

diameter origin realm

To configure origin realm information for a Diameter node, use the **diameter origin realm** command in global configuration mode. To disable the configured realm information, use the **no** form of this command.

diameter origin realm *string*

no diameter origin realm *string*

Syntax Description

string	Character string that describes the realm information for a specific Diameter node.

Command Default

No realm is configured.

Command Modes

Global configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

Because there is no realm configured by default, it is mandatory to configure this information. Origin realm information is sent in requests to a Diameter peer.

Examples

The following example shows how to configure a Diameter origin realm:

```
Router (config)# diameter origin realm example.com
```

Related Commands

Command	Description
diameter origin host	Configures the FQDN of the host of a Diameter node.
diameter peer	Defines a Diameter peer and enters Diameter peer configuration mode.

diameter peer

To configure a device as a Diameter Protocol peer and enter the Diameter peer configuration submode, use the **diameter peer** command in global configuration mode. To disable Diameter Protocol configuration for a peer, use the **no** form of this command.

diameter peer *name*

no diameter peer *name*

Syntax Description

name	Character string used to name the peer node to be configured for the Diameter Credit Control Application (DCCA).

Command Default

No Diameter peer is configured.

Command Modes

Global configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

This command enables the Diameter peer configuration submode. From the submode, you can configure other DCCA parameters. The configuration is applied when you exit the submode.

Examples

The following example shows how to configure a Diameter peer:

```
Router (config)# diameter peer dia_peer_1
```

Related Commands

Command	Description
address ipv4	Defines a route to the host of the Diameter peer using IPv4.
destination host	Configures the FQDN of a Diameter peer.
destination realm	Configures the destination realm in which a Diameter peer is located.
ip vrf forwarding	Associates a VRF with a Diameter peer.
security ipsec	Configures IPSec as the security protocol for the Diameter peer-to-peer connection.
show diameter peer	Displays the Diameter peer configuration.
source interface	Configures the interface to connect to the Diameter peer.
timer	Configures Diameter base protocol timers for peer-to-peer communication.
transport {tcp} port	Configures the transport protocol for connections to the Diameter peer.

diameter redundancy

To enable the Diameter node to be a Cisco IOS Redundancy Facility (RF) client and track session states, use the **diameter redundancy** command in global configuration mode. To disable this feature, use the **no** form of this command.

diameter redundancy

no diameter redundancy

Syntax Description

This command has no arguments or keywords.

Command Default

Diameter redundancy is not configured.

Command Modes

Global configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

When you configure Diameter redundancy on a device, that device will not initiate any TCP connection while it is a standby node. Upon transition to active status, the device initiates a TCP connection to the Diameter peer.

Note This command is required for service-aware Packet Data Protocol (PDP) session redundancy. For more information about service-aware PDP session redundancy, see the "GTP-Session Redundancy for Service-Aware PDPs Overview" section of the *Cisco GGSN Release 5.2 Configuration Guide*.

Examples

The following example shows how to configure Diameter redundancy:

```
Router (config)# diameter redundancy
```

Related Commands

Command	Description
diameter origin host	Configures the FQDN of the host of this Diameter node.
diameter origin realm	Configures the realm of origin in which this Diameter node is located.
diameter timer	Configures Diameter base protocol timers to use if none have been configured at the Diameter peer level.
diameter vendor support	Configures a Diameter node to advertise the vendor AVPs it supports in capability exchange messages with Diameter peers.

diameter timer

To set either the frequency of transport connection attempts or the interval for sending watchdog messages, use the **diameter timer** command in global configuration mode. To return to the default values, use the **no** form of this command.

diameter timer {**connection** | **transaction** | **watch-dog**} *value*

no diameter timer {**connection** | **transaction** | **watch-dog**} *value*

Syntax Description

connection	Maximum interval, in seconds, for the Gateway General Packet Radio Service (GPRS) Support Node (GGSN) to attempt reconnection to a Diameter peer after being disconnected due to a transport failure. The range is from 1 to 1000. The default is 30. A value of 0 configures the GGSN not to attempt reconnection.
transaction	Maximum interval, in seconds, the GGSN waits for a Diameter peer to respond before trying another peer. The range is from 1 to 1000. The default is 30.
watch-dog	Maximum interval, in seconds, the GGSN waits for a Diameter peer response to a watchdog packet. The range is from 1 to 1000. The default is 30. **Note** When the watchdog timer expires, a device watchdog request (DWR) is sent to the Diameter peer and the watchdog timer is reset. If a device watchdog answer (DWA) is not received before the next expiration of the watchdog timer, a transport failure to the Diameter peer has occurred.
value	The valid range, in seconds, from 1 to 1000. The default is 30.

Command Default

The default value for each timer is 30 seconds.

Command Modes

Global configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

When configuring timers, the value for the transaction timer should be larger than the transmission-timeout value, and, on the Serving GPRS Support Node (SGSN), the values configured for the number of GPRS Tunneling Protocol (GTP) N3 requests and T3 retransmissions must be larger than the sum of all possible server timers (RADIUS, Diameter Credit Control Application (DCCA), and Cisco Content Services Gateway (CSG)). Specifically, the SGSN N3*T3 must be greater than 2 x RADIUS timeout + *N* x DCCA timeout + CSG timeout where:

- The factor *2* is for both authentication and accounting.
- The value *N* is for the number of Diameter servers configured in the server group.

Examples

The following examples show how to configure the Diameter timers:

```
Router config# diameter timer connection 20

Router config# diameter timer watch-dog 25
```

Related Commands

Command	Description
aaa group server diameter	Defines a Diameter AAA server group.
diameter peer	Configures a Diameter peer and enters Diameter peer configuration submode.
timer	Configures the Diameter base protocol timers for a Diameter peer.

diameter vendor supported

To configure a Diameter node to advertise the vendor-specific attribute value pairs (AVPs) it recognizes, use the **diameter vendor supported** command in global configuration mode. To remove the supported vendor configuration, use the **no** form of this command.

diameter vendor supported {**Cisco | 3gpp | Vodafone**}

no diameter vendor supported {**Cisco | 3gpp | Vodafone**}

Syntax Description

Cisco	Configures the Diameter node to advertise support for the Cisco-specific AVPs.
3gpp	Configures the Diameter node to advertise support for the AVPs that support the Third-Generation Partnership Project (3GPP).
Vodafone	Configures the Diameter node to advertise support for the Vodafone-specific AVPs.

Command Default

No vendor identifier is configured.

Command Modes

Global configuration

Command History

Release	Modification
12.4(9)T	This command was introduced.

Usage Guidelines

Individual vendors can define AVPs specific to their implementation of the Diameter Credit Control Application (DCCA), or for individual applications. You can configure multiple instances of this command, as long as each instance has a different vendor identifier.

Examples

The following example shows how to configure DCCA to advertise support for a the Cisco-specific AVPs:

```
Router (config)# diameter vendor supported Cisco
```

Related Commands

Command	Description
diameter origin host	Configures the FQDN of the host of this Diameter node.
diameter origin realm	Configures the realm of origin in which this Diameter node is located.
diameter redundancy	Enables the Diameter node to be a Cisco IOS RF client and track session states.
diameter timer	Configures Diameter base protocol timers to use if none have been configured at the Diameter peer level.

disable open-media-channel

To prevent the creation of Real-time Transport Protocol (RTP) or RTP Control (RTCP) media channels when a Session Initiation Protocol (SIP) class map is used for SIP inspection, use the **disable open-media-channel** command in parameter-map type configuration mode. To enable the creation of RTP or RTCP media channels, use the **no** form of this command or remove this parameter map from the inspect action.

disable open-media-channel

no disable open-media-channel

Syntax Description

This command has no arguments or keywords.

Command Default

RTP and RTPC media channels are opened by the SIP inspection process.

Command Modes

Parameter-map type configuration (config-profile)

Command History

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

Usage Guidelines

Cisco IOS Firewall Trust Relay Point (TRP) support enables Cisco IOS Firewall to process Simple Traversal of User Datagram Protocol (UDP) (STUN) messages. The STUN messages open ports (pinholes) for secondary channels (RTP and RTCP), which are necessary for implementation of TRPs in voice networks.

Cisco IOS Firewall supports partial SIP inspection that allows the SIP Application-level Gateway (ALG) to parse the SIP message in a packet to check for protocol conformance.

To configure partial SIP inspection in voice networks, you must use the **disable open-media-channel** command to configure SIP ALG so that it does not open pinholes for media information found in the SDP message.

When Cisco IOS TRP is used in voice network for firewall traversal, Partial SIP-ALG (enabled when this parameter map is attached to the inspect action) provides security for SIP control channel and STUN with Cisco Flow data (CFD) provides security for the RTP and RTCP channels. If Partial SIP-ALG is not used, the normal SIP-ALG will open RTP and RTCP channels by itself.

Examples

The following example shows how to create a parameter map that does not open a media channel when attached to a SIP class map:

```
Router(config)# parameter-map type protocol-info sip pmap-sip
Router(config-profile)# disable open-media-channel
```

Related Commands

Command	Description
parameter-map type protocol-info	Creates or modifies a protocol-specific parameter map and enters parameter-map type configuration mode.

disconnect ssh

To terminate a Secure Shell (SSH) connection on your router, use the **disconnect ssh** command in privileged EXEC mode.

disconnect ssh [**vty**] *session-id*

Syntax Description

vty	(Optional) Virtual terminal for remote console access.
session-id	The session-id is the number of connection displayed in the **show ip ssh** command output.

Defaults

No default behavior or values.

Command Modes

Privileged EXEC

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 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.

Usage Guidelines

The **clear line vty** *n* command, where *n* is the connection number displayed in the **show ip ssh** command output, may be used instead of the **disconnect ssh** command.

When the EXEC connection ends, whether normally or abnormally, the SSH connection also ends.

Examples

The following example terminates SSH connection number 1:

```
disconnect ssh 1
```

Related Commands

Command	Description
clear line vty	Returns a terminal line to idle state using the privileged EXEC command.

dn

To associate the identity of a router with the distinguished name (DN) in the certificate of the router, use the **dn** command in crypto identity configuration mode. To remove this command from your configuration, use the **no** form of this command.

dn *name=string* [, *name=string*]

no dn *name=string* [, *name=string*]

Syntax Description

name=string	Identity used to restrict access to peers with specific certificates. Optionally, you can associate more than one identity.

Command Default

If this command is not enabled, the router can communicate with any encrypted interface that is not restricted on its IP address.

Command Modes

Crypto identity configuration

Command History

Release	Modification
12.2(4)T	This command was introduced.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(33)SRA.

Usage Guidelines

Use the **dn** command to associate the identity of the router, which is defined in the **crypto identity** command, with the DN that the peer used to authenticate itself.

Note The *name* defined in the **crypto identity** command must match the *string* defined in the **dn** command. That is, the identity of the peer must be the same as the identity in the exchanged certificate.

This command allows you set restrictions in the router configuration that prevent those peers with specific certificates, especially certificates with particular DNs, from having access to selected encrypted interfaces.

An encrypting peer matches this list if it contains the attributes listed in any one line defined within the *name=string*.

Examples

The following example shows how to configure an IPsec crypto map that can be used only by peers that have been authenticated by the DN and if the certificate belongs to “green”:

```
crypto map map-to-green 10 ipsec-isakmp
 set peer 172.21.114.196
 set transform-set my-transformset
 match address 124
 identity to-green
!
crypto identity to-green
 dn ou=green
```

Related Commands

Command	Description
crypto identity	Configures the identity of the router with a given list of DNs in the certificate of the router.
fqdn	Associates the identity of the router with the hostname that the peer used to authenticate itself.

dn (IKEv2)

To enable and derive an IKEv2 name mangler from identity of type distinguished name (DN), use the **dn** command in IKEv2 name mangler configuration mode. To remove the name derived from DN, use the **no** form of this command.

dn {**common-name** | **country** | **domain** | **locality** | **organization** | **organization-unit** | **state**}

no dn

Syntax Description

common-name	Derives the name mangler from the common name portion in the DN.
country	Derives the name mangler from the country portion in the DN.
domain	Derives the name mangler from the domain portion in the DN.
locality	Derives the name mangler from the locality portion in the DN.
organization	Derives the name mangler from the organization portion in the DN.
organization-unit	Derives the name mangler from the organization-unit portion in the DN.
state	Derives the name mangler from the state portion in the DN.

Command Default

No default behavior or values.

Command Modes

IKEv2 name mangler configuration (config-ikev2-name-mangler)

Command History

Release	Modification
15.1(3)T	This command was introduced.
Cisco IOS XE Release 3.3S	This command was integrated into Cisco IOS XE Release 3.3S.

Usage Guidelines

Use this command to derive the name mangler from any field in the remote identity of type DN.

Examples

The following example shows how to derive a name for the name mangler from the country field of the DN:

```
Router(config)# crypto ikev2 name-mangler mangler2
Router(config-ikev2-name-mangler)# dn country
```

Related Commands

Command	Description
crypto ikev2 name mangler	Defines a name mangler.

dnis (AAA preauthentication)

To preauthenticate calls on the basis of the Dialed Number Identification Service (DNIS) number, use the **dnis** command in AAA preauthentication configuration mode. To remove the **dnis** command from your configuration, use the **no** form of this command.

dnis [**if-avail | required**] [**accept-stop**] [**password** *string*]

no dnis [**if-avail | required**] [**accept-stop**] [**password** *string*]

Syntax Description

if-avail	(Optional) Implies that if the switch provides the data, RADIUS must be reachable and must accept the string in order for preauthentication to pass. If the switch does not provide the data, preauthentication passes.
required	(Optional) Implies that the switch must provide the associated data, that RADIUS must be reachable, and that RADIUS must accept the string in order for preauthentication to pass. If these three conditions are not met, preauthentication fails.
accept-stop	(Optional) Prevents subsequent preauthentication elements from being tried once preauthentication has succeeded for a call element.
password *string*	(Optional) Password to use in the Access-Request packet. The default is cisco.

Defaults

The **if-avail** and **required** keywords are mutually exclusive. If the **if-avail** keyword is not configured, the preauthentication setting defaults to **required**.

The default password string is cisco.

Command Modes

AAA preauthentication configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.

Usage Guidelines

You may configure more than one of the AAA preauthentication commands (**clid**, **ctype**, **dnis**) to set conditions for preauthentication. The sequence of the command configuration decides the sequence of the preauthentication conditions. For example, if you configure **dnis**, then **clid**, then **ctype**, then this is the order of the conditions considered in the preauthentication process.

In addition to using the preauthentication commands to configure preauthentication on the Cisco router, you must set up the preauthentication profiles on the RADIUS server.

Examples

The following example enables DNIS preauthentication using a RADIUS server and the password Ascend-DNIS:

```
aaa preauth
 group radius
 dnis password Ascend-DNIS
```

The following example specifies that incoming calls be preauthenticated on the basis of the DNIS number:

```
aaa preauth
 group radius
 dnis required
```

Related Commands

Command	Description
aaa preauth	Enters AAA preauthentication mode.
clid	Preauthenticates calls on the basis of the CLID number.
ctype	Preauthenticates calls on the basis of the call type.
dnis bypass (AAA preauthentication configuration)	Specifies a group of DNIS numbers that will be bypassed for preauthentication.
group (authentication)	Selects the security server to use for AAA preauthentication.
isdn guard-timer	Sets a guard timer to accept or reject a call in the event that the RADIUS server fails to respond to a preauthentication request.

dnis (RADIUS)

To preauthenticate calls on the basis of the DNIS (Dialed Number Identification Service) number, use the **dnis** command in AAA preauthentication configuration mode. To remove the **dnis** command from your configuration, use the **no** form of this command.

dnis [**if-avail | required**] [**accept-stop**] [**password** *password*]

no dnis [**if-avail | required**] [**accept-stop**] [**password** *password*]

Syntax Description

if-avail	(Optional) Implies that if the switch provides the data, RADIUS must be reachable and must accept the string in order for preauthentication to pass. If the switch does not provide the data, preauthentication passes.
required	(Optional) Implies that the switch must provide the associated data, that RADIUS must be reachable, and that RADIUS must accept the string in order for preauthentication to pass. If these three conditions are not met, preauthentication fails.
accept-stop	(Optional) Prevents subsequent preauthentication elements such as **clid** or **ctype** from being tried once preauthentication has succeeded for a call element.
password *password*	(Optional) Defines the password for the preauthentication element.

Defaults

The **if-avail** and **required** keywords are mutually exclusive. If the **if-avail** keyword is not configured, the preauthentication setting defaults to **required**.

The default password string is cisco.

Command Modes

AAA preauthentication configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

You may configure more than one of the authentication, authorization, and accounting (AAA) preauthentication commands (**clid**, **ctype**, **dnis**) to set conditions for preauthentication. The sequence of the command configuration decides the sequence of the preauthentication conditions. For example, if you configure **dnis**, then **clid**, then **ctype**, in this order, then this is the order of the conditions considered in the preauthentication process.

In addition to using the preauthentication commands to configure preauthentication on the Cisco router, you must set up the preauthentication profiles on the RADIUS server.

Examples

The following example specifies that incoming calls be preauthenticated on the basis of the DNIS number:

```
aaa preauth
 group radius
 dnis required
```

Related Commands

Command	Description
clid	Preauthenticates calls on the basis of the CLID number.
ctype	Preauthenticates calls on the basis of the call type.
dnis bypass (AAA preauthentication configuration)	Specifies a group of DNIS numbers that will be bypassed for preauthentication.
group (RADIUS)	Specifies the AAA RADIUS server group to use for preauthentication.

dnis bypass (AAA preauthentication configuration)

To specify a group of DNIS (Dialed Number Identification Service) numbers that will be bypassed for preauthentication, use the **dnis bypass** command in AAA preauthentication configuration mode. To remove the **dnis bypass** command from your configuration, use the **no** form of this command.

dnis bypass {*dnis-group-name*}

no dnis bypass {*dnis-group-name*}

Syntax Description

dnis-group-name	Name of the defined DNIS group.

Defaults

No DNIS numbers are bypassed for preauthentication.

Command Modes

AAA preauthentication configuration

Command History

Release	Modification
12.1(2)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS release 12.(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Before using this command, you must first create a DNIS group with the **dialer dnis group** command.

Examples

The following example specifies that preauthentication be performed on all DNIS numbers except for two DNIS numbers (12345 and 12346), which have been defined in the DNIS group called hawaii:

```
aaa preauth
 group radius
 dnis required
 dnis bypass hawaii

dialer dnis group hawaii
 number 12345
 number 12346
```

Related Commands

Command	Description
dialer dnis group	Creates a DNIS group.
dnis (RADIUS)	Preauthenticates calls on the basis of the DNIS number.

dns

To specify the primary and secondary Domain Name Service (DNS) servers, use the **dns** command in ISAKMP group configuration mode or IKEv2 authorization policy configuration mode. To remove this command from your configuration, use the **no** form of this command.

dns *primary-server* [*secondary-server*]

no dns *primary-server* [*secondary-server*]

Syntax Description

primary-server	Name of the primary DNS server.
secondary-server	(Optional) Name of the secondary DNS server.

Defaults

A DNS server is not specified.

Command Modes

ISAKMP group configuration (config-isakmp-group)
IKEv2 authorization policy configuration (config-ikev2-author-policy)

Command History

Release	Modification
12.2(8)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 3.3S	This command was integrated into Cisco IOS XE Release 3.3S.

Usage Guidelines

Use the **dns** command to specify the primary and secondary DNS servers for the group.

You must enable the following commands before enabling the **dns** command:

- **crypto isakmp client configuration group**—Specifies the group policy information that has to be defined or changed.
- **crypto ikev2 authorization policy**—Specifies the local group policy authorization parameters.

Examples

The following example shows how to define a primary and secondary DNS server for the default group name:

```
crypto isakmp client configuration group default
 key cisco
 dns 10.2.2.2 10.3.2.3
 pool dog
 acl 199
```

Related Commands

Command	Description
acl	Configures split tunneling.
crypto ikev2 authorization policy	Specifies an IKEv2 authorization policy.
crypto isakmp client configuration group	Specifies the policy profile of the group that will be defined.
domain (isakmp-group)	Specifies the DNS domain to which a group belongs.