

ip trigger-authentication (global)

To enable the automated part of double authentication at a device, use the **ip trigger-authentication** command in global configuration mode. To disable the automated part of double authentication, use the **no** form of this command.

ip trigger-authentication [**timeout** *seconds*] [**port** *number*]

no ip trigger-authentication

Syntax Description

timeout <i>seconds</i>	(Optional) Specifies how frequently the local device sends a User Datagram Protocol (UDP) packet to the remote host to request the user's username and password (or PIN). The default is 90 seconds. See "The Timeout Keyword" in the Usage Guidelines section for details.
port <i>number</i>	(Optional) Specifies the UDP port to which the local router should send the UPD packet requesting the user's username and password (or PIN). The default is port 7500. See "The Port Keyword" in the Usage Guidelines section for details.

Defaults

The default timeout is 90 seconds, and the default port number is 7500.

Command Modes

Global configuration

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

Configure this command on the local device (router or network access server) that remote users dial in to. Use this command only if the local device has already been configured to provide double authentication; this command enables automation of the second authentication of double authentication.

The timeout Keyword

During the second authentication stage of double authentication—when the remote user is authenticated—the remote user must send a username and password (or PIN) to the local device. With automated double authentication, the local device sends a UDP packet to the remote user's host during the second user-authentication stage. This UDP packet triggers the remote host to launch a dialog box requesting a username and password (or PIN).

If the local device does not receive a valid response to the UDP packet within a timeout period, the local device will send another UDP packet. The device will continue to send UDP packets at the timeout intervals until it receives a response and can authenticate the user.

By default, the UDP packet timeout interval is 90 seconds. Use the **timeout** keyword to specify a different interval.

(This timeout also applies to how long entries will remain in the remote host table; see the **show ip trigger-authentication** command for details.)

The port Keyword

As described in the previous section, the local device sends a UDP packet to the remote user's host to request the user's username and password (or PIN). This UDP packet is sent to UDP port 7500 by default. (The remote host client software listens to UDP port 7500 by default.) If you need to change the port number because port 7500 is used by another application, you should change the port number using the **port** keyword. If you change the port number you need to change it in both places—both on the local device and in the remote host client software.

Examples

The following example globally enables automated double authentication and sets the timeout to 120 seconds:

```
ip trigger-authentication timeout 120
```

Related Commands

Command	Description
ip trigger-authentication (interface)	Specifies automated double authentication at an interface.
show ip trigger-authentication	Displays the list of remote hosts for which automated double authentication has been attempted.

ip trigger-authentication (interface)

To specify automated double authentication at an interface, use the **ip trigger-authentication** command in interface configuration mode. To turn off automated double authentication at an interface, use the **no** form of this command.

ip trigger-authentication

no ip trigger-authentication

Syntax Description This command has no arguments or keywords.

Defaults Automated double authentication is not enabled for specific interfaces.

Command Modes Interface configuration

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines Configure this command on the local router or network access server that remote users dial into. Use this command only if the local device has already been configured to provide double authentication and if automated double authentication has been enabled with the **ip trigger-authentication** (global) command.

This command causes double authentication to occur automatically when users dial into the interface.

Examples The following example turns on automated double authentication at the ISDN BRI interface BRI0:

```
interface BRI0
 ip trigger-authentication
 encapsulation ppp
 ppp authentication chap
```

Related Commands	Command	Description
	ip trigger-authentication (global)	Enables the automated part of double authentication at a device.

ip urlfilter alert

To enable URL filtering system alert messages, use the **ip urlfilter alert** command in global configuration mode. To disable the system alert, use the **no** form of this command.

ip urlfilter alert

no ip urlfilter alert

Syntax Description This command has no arguments or keywords.

Defaults URL filtering messages are enabled.

Command Modes Global configuration

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines Use the **ip urlfilter alert** command to display system messages, such as a server entering allow mode, a server going down, or a URL that is too long for the lookup request.

Examples The following example shows how to enable URL filtering alert messages:

```
ip inspect name test http urlfilter
ip urlfilter cache 5
ip urlfilter exclusive-domain permit .weapons.com
ip urlfilter exclusive-domain deny .nbc.com
ip urlfilter exclusive-domain permit www.cisco.com
ip urlfilter audit-trail
ip urlfilter alert
ip urlfilter server vendor websense 192.168.3.1
```

Afterward, system alert messages such as the following are displayed:

```
%URLF-3-SERVER_DOWN:Connection to the URL filter server 10.92.0.9 is down
```

This level three LOG_ERR-type message is displayed when a configured URL filter server (UFS) goes down. When this happens, the firewall will mark the configured server as secondary and try to bring up one of the other secondary servers and mark that server as the primary server. If there is no other server configured, the firewall will enter into allow mode and display the URLF-3-ALLOW_MODE message described.

```
%URLF-3-ALLOW_MODE:Connection to all URL filter servers are down and ALLOW MODE is OFF
```

This LOG_ERR type message is displayed when all UFSs are down and the system enters into allow mode.



Note Whenever the system goes into allow mode (all filter servers are down), a periodic keepalive timer will be triggered that will try to bring up a server by opening a TCP connection.

```
%URLF-5-SERVER_UP:Connection to an URL filter server 10.92.0.9 is made, the system is returning from ALLOW MODE
```

This LOG_NOTICE-type message is displayed when the UFSs are detected as being up and the system is returning from allow mode.

```
%URLF-4-URL_TOO_LONG:URL too long (more than 3072 bytes), possibly a fake packet?
```

This LOG_WARNING-type message is displayed when the URL in a lookup request is too long; any URL longer than 3K will be dropped.

```
%URLF-4-MAX_REQ:The number of pending request exceeds the maximum limit <1000>
```

This LOG_WARNING-type message is displayed when the number of pending requests in the system exceeds the maximum limit and all further requests are dropped.

ip urlfilter allowmode

To turn on the default mode (allow mode) of the filtering algorithm, use the **ip urlfilter allowmode** command in global configuration mode. To disable the default mode, use the **no** form of this command.

ip urlfilter allowmode [on | off]

no ip urlfilter allowmode [on | off]

Syntax Description	on	(Optional) Allow mode is on.
	off	(Optional) Allow mode is off.

Defaults Allow mode is off.

Command Modes Global configuration

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines The system will go into allow mode when connections to all vendor servers (Websense or N2H2) are down. The system will return to normal mode when a connection to at least one web vendor server is up. Allow mode directs your system to forward or drop all packets on the basis of the configurable allow mode setting; if allow mode is on and the vendor servers are down, the HTTP requests will be allowed to pass; if allow mode is off and the vendor servers are down, the HTTP requests will be forbidden.

Examples The following example shows how to enable allow mode on your system:

```
ip urlfilter allowmode on
```

Afterward, the following alert message will be displayed when the system goes into allow mode:

```
%URLF-3-ALLOW_MODE: Connection to all URL filter servers are down and ALLOW MODE if OFF
```

The following alert message will be displayed when the system returns from allow mode:

```
%URLF-5-SERVER_UP: Connection to an URL filter server 12.0.0.3 is made, the system is returning from allow mode
```

ip urlfilter audit-trail

To log messages into the syslog server or router, use the **ip urlfilter audit-trail** command in global configuration mode. To disable this functionality, use the **no** form of this command.

ip urlfilter audit-trail

no ip urlfilter audit-trail

Syntax Description This command has no arguments or keywords.

Defaults This command is disabled.

Command Modes Global configuration

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines Use the **ip urlfilter audit-trail** command to log messages such as URL request status (allow or deny) into your syslog server.

Examples The following example shows how to enable syslog message logging:

```
ip inspect name test http urlfilter
ip urlfilter cache 5
ip urlfilter exclusive-domain permit .weapons.com
ip urlfilter exclusive-domain deny .nbc.com
ip urlfilter exclusive-domain permit www.cisco.com
ip urlfilter audit-trail
ip urlfilter alert
ip urlfilter server vendor websense 209.165.202.130
```

Afterward, audit trail messages such as the following are displayed and logged into the log server:

```
%URLF-6-SITE_ALLOWED:Client 209.165.201.15:12543 accessed server 10.76.82.21:8080
```

This message is logged for each request whose destination IP address is found in the cache. It includes the source IP address, source port number, destination IP address, and destination port number. The URL is not logged in this case because the IP address of the request is found in the cache; thus, parsing the request and extracting the URL is a waste of time.

```
%URLF-4-SITE-BLOCKED: Access denied for the site 'www.sports.com'; client  
209.165.200.230:34557 server 209.165.201.2:80
```

This message is logged when a request finds a match against one of the blocked domains in the exclusive-domain list or the corresponding entry in the IP cache.

```
%URLF-6-URL_ALLOWED:Access allowed for URL http://www.N2H2.com/; client  
209.165.200.230:54123 server 192.168.0.1:80
```

This message is logged for each URL request that is allowed by the vendor server (Websense or N2H2). It includes the allowed URL, source IP address, source port number, destination IP address, and destination port number. Longer URLs will be truncated to 300 bytes and then logged.

```
%URLF-6-URL_BLOCKED:Access denied URL http://www.google.com; client 209.165.200.230:54678  
server 209.165.201.2:80
```

This message is logged for each URL request that is blocked by the vendor server. It includes the blocked URL, source IP address, source port number, destination IP address, and destination port number. Longer URLs will be truncated to 300 bytes and then logged.

ip urlfilter cache

To configure cache parameters, use the **ip urlfilter cache** command in global configuration mode. To clear the configuration, use the **no** form of this command.

ip urlfilter cache *number*

no ip urlfilter cache *number*

Syntax Description	<i>number</i>	Maximum number of destination IP addresses that can be cached into the cache table. The default value is 5000.
--------------------	---------------	--

Defaults	Maximum number of destination IP addresses is 5000. The cache table is cleared out every 12 hours.
----------	---

Command Modes	Global configuration
---------------	----------------------

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines	The cache table consists of the most recently requested IP addresses and respective authorization status for each IP address.
------------------	---

The caching algorithm involves three parameters—the maximum number of IP addresses that can be cached, an idle time, and an absolute time. The algorithm also involves two timers—idle timer and absolute timer. The idle timer is a small periodic timer (1 minute) that checks to see whether the number of cached IP addresses in the cache table exceeds 80 percent of the maximum limit. If the cached IP addresses have exceeded 80 percent, it will start removing idle entries; if it has not exceeded 80 percent, it will quit and wait for the next cycle. The absolute timer is a large periodic timer (1 hour) that is used to remove all of the elapsed entries. (The age of an elapsed entry is greater than the absolute time.) An elapsed entry will also be removed during cache lookup.

The idle time value is fixed at 10 minutes. The absolute time value is taken from the vendor server look-up response, which is often greater than 15 hours. The absolute value for cache entries made out of exclusive-domains is 12 hours. The maximum number of cache entries is configurable by enabling the **ip urlfilter cache** command.



Note	The vendor server is not able to inform the Cisco IOS firewall of filtering policy changes in the database.
------	---

Examples

The following example shows how to configure the cache table to hold a maximum of five destination IP addresses:

```
ip inspect name test http urlfilter
ip urlfilter cache 5
ip urlfilter exclusive-domain permit .weapons.com
ip urlfilter exclusive-domain deny .nbc.com
ip urlfilter exclusive-domain permit www.cisco.com
ip urlfilter audit-trail
ip urlfilter alert
ip urlfilter server vendor websense 192.168.3.1
```

Related Commands

Command	Description
clear ip urlfilter cache	Clears the cache table.
show ip urlfilter cache	Displays the destination IP addresses that are cached into the cache table.

ip urlfilter exclusive-domain

To add or remove a domain name to or from the exclusive domain list so that the firewall does not have to send lookup requests to the vendor server, use the **ip urlfilter exclusive-domain** command in global configuration mode. To remove a domain name from the exclusive domain name list, use the **no** form of this command.

ip urlfilter exclusive-domain { **permit** | **deny** } *domain-name*

no ip urlfilter exclusive-domain { **permit** | **deny** } *domain-name*

Syntax Description		
	permit	Permits all traffic destined for the specified domain name.
	deny	Blocks all traffic destined for the specified domain name.
	<i>domain-name</i>	Domain name that is added or removed from the exclusive domain name list; for example, www.cisco.com.

Defaults This command is not enabled.

Command Modes Global configuration

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines The **ip urlfilter exclusive-domain** command allows you to specify a list of domain names (exclusive domains) so that the firewall will not create a lookup request for the HTTP traffic that is destined for one of the domains in the exclusive list. Thus, you can avoid sending look-up requests to the web server for HTTP traffic that is destined for a host that is completely allowed to all users.

Flexibility when entering domain names is also provided; that is, the user can enter the complete domain name or a partial domain name.

Complete Domain Name

If the user adds a complete domain name, such as “www.cisco.com,” to the exclusive domain list, all HTTP traffic whose URLs are destined for this domain (such as www.cisco.com/news and www.cisco.com/index) will be excluded from the URL filtering policies of the vendor server (Websense or N2H2), and on the basis of the configuration, the URLs will be permitted or blocked (denied).

Partial Domain Name

If the user adds only a partial domain name to the exclusive domain list, such as “.cisco.com,” all URLs whose domain names end with this partial domain name (such as www.cisco.com/products and www.cisco.com/eng) will be excluded from the URL filtering policies of the vendor server (Websense or N2H2), and on the basis of the configuration, the URLs will be permitted or blocked (denied).

Examples

The following example shows how to add the complete domain name “www.cisco.com” to the exclusive domain name list. This configuration will block all traffic destined to the www.cisco.com domain.

```
ip urlfilter exclusive-domain deny www.cisco.com
```

The following example shows how to add the partial domain name “.cisco.com” to the exclusive domain name list. This configuration will permit all traffic destined to domains that end with .cisco.com.

```
ip urlfilter exclusive-domain permit .cisco.com
```

ip urlfilter max-request

To set the maximum number of outstanding requests that can exist at any given time, use the **ip urlfilter max-request** command in global configuration mode. To disable this function, use the **no** form of this command.

ip urlfilter max-request *number*

no ip urlfilter max-request *number*

Syntax Description	<i>number</i>	Maximum number of outstanding requests. The default value is 1000.
--------------------	---------------	--

Defaults	Maximum number of requests is 1000.
----------	-------------------------------------

Command Modes	Global configuration
---------------	----------------------

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.	

Usage Guidelines	If the specified maximum number of outstanding requests is exceeded, new requests will be dropped.
------------------	--



Note

Allow mode is not considered because it should be used only when servers are down.
--

Examples	The following example shows how to configure the maximum number of outstanding requests to 950:
----------	---

```
ip inspect name url_filter http
ip urlfilter max-request 950
```

Related Commands	Command	Description
	ip inspect name	Defines a set of inspection rules.
ip urlfilter server vendor	Configures a vendor server for URL filtering.	

ip urlfilter max-resp-pak

To configure the maximum number of HTTP responses that the firewall can keep in its packet buffer, use the **ip urlfilter max-resp-pak** command in global configuration mode. To return to the default, use the **no** form of this command.

ip urlfilter max-resp-pak *number*

no ip urlfilter max-resp-pak *number*

Syntax Description	<i>number</i>	Maximum number of HTTP responses that can be stored in the packet buffer of the firewall. After the maximum number has been reached, the firewall will drop further responses. The default, and absolute maximum, value is 200.
---------------------------	---------------	---

Defaults	200 HTTP responses
-----------------	--------------------

Command Modes	Global configuration
----------------------	----------------------

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.	

Usage Guidelines

When an HTTP request arrives at a Cisco IOS firewall, the firewall forwards the request to the web server while simultaneously sending a URL look-up request to the vendor server (Websense or N2H2). If the vendor server reply arrives before the HTTP response, the firewall will know whether to permit or block the HTTP response; if the HTTP response arrives before the vendor server reply, the firewall will not know whether to allow or block the response, so the firewall will drop the response until it hears from the vendor server. The **ip urlfilter max-resp-pak** command allows you to configure your firewall to store the HTTP responses in a buffer, which allows your firewall to store a maximum of 200 HTTP responses. Each response will remain in the buffer until an allow or deny message is received from the vendor server. If the vendor server reply allows the URL, the firewall will release the HTTP response from the buffer to the end user; if the vendor server reply denies the URL, the firewall will discard the HTTP response from the buffer and close the connection to both ends.

Examples

The following example shows how to configure your firewall to hold 150 HTTP responses:

```
ip urlfilter max-resp-pak 150
```

ip urlfilter server vendor

To configure a vendor server for URL filtering, use the **ip urlfilter server vendor** command in global configuration mode. To remove a server from your configuration, use the **no** form of this command.

```
ip urlfilter server vendor { websense | n2h2 } ip-address [port port-number] [timeout seconds]
[retransmit number]
```

```
no ip urlfilter server vendor { websense | n2h2 } ip-address [port port-number] [timeout seconds]
[retransmit number]
```

Syntax Description		
websense		Websense server will be used.
n2h2		N2H2 server will be used.
<i>ip-address</i>		IP address of the vendor server.
port <i>port-number</i>		(Optional) Port number that the vendor server listens on. The default port number is 15868.
timeout <i>seconds</i>		(Optional) Length of time, in seconds, that the Cisco IOS firewall will wait for a response from the vendor server. The default timeout is 5 seconds.
retransmit <i>number</i>		(Optional) Number of times the Cisco IOS firewall will retransmit the request when a response does not arrive for the request. The default value is two times.

Defaults A vendor server is not configured.

Command Modes Global configuration

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines Use the **ip urlfilter server vendor** command to configure a Websense or N2H2 server, which will interact with the Cisco IOS Firewall to filter HTTP requests on the basis of a specified policy—global filtering, user- or group-based filtering, keyword-based filtering, category-based filtering, or customized filtering.

If the firewall has not received a response from the vendor server within the time specified in the **timeout seconds** keyword and argument, the firewall will check the **retransmit number** keyword and argument configured for the vendor server. If the firewall *has not* exceeded the maximum retransmit tries allowed, it will resend the HTTP lookup request. If the firewall *has* exceeded the maximum retransmit tries allowed, it will delete the outstanding request from the queue and check the status of the allow mode value. The firewall will forward the request if the allow mode is on; otherwise, it will drop the request.

Primary and Secondary Servers

When users configure multiple vendor servers, the firewall will use only one server at a time—the primary server; all other servers are called secondary servers. When the primary server becomes unavailable for any reason, it becomes a secondary server and one of the secondary servers becomes the primary server.

A firewall marks a primary server as down when sending a request to or receiving a response from the server fails. When a primary server goes down, the system will go to the beginning of the configured servers list and try to activate the first server on the list. If the first server on the list is unavailable, it will try the second server on the list; the system will keep trying to activate a server until it is successful or until it reaches the end of the server list. If the system reaches the end of the server list, it will set a flag indicating that all of the servers are down, and it will enter allow mode.

Examples

The following example shows how to configure the Websense server for URL filtering:

```
ip inspect name test http urlfilter
ip urlfilter cache 5
ip urlfilter exclusive-domain permit .weapons.com
ip urlfilter exclusive-domain deny .nbc.com
ip urlfilter exclusive-domain permit www.cisco.com
ip urlfilter audit-trail
ip urlfilter alert
ip urlfilter server vendor websense 192.168.3.1
```

Related Commands

Command	Description
ip urlfilter allowmode	Turns on the default mode (allow mode) of the filtering algorithm.
ip urlfilter max-request	Sets the maximum number of outstanding requests that can exist at any given time.

ip urlfilter urlf-server-log

To enable the logging of system messages on the URL filtering server, use the **ip urlfilter urlf-server-log** command in global configuration mode. To disable the logging of system messages, use the **no** form of this command.

ip urlfilter urlf-server-log

no ip urlfilter urlf-server-log

Syntax Description This command has no arguments or keywords.

Defaults This command is disabled.

Command Modes Global configuration

Command History	Release	Modification
	12.2(11)YU	This command was introduced.
	12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.

Usage Guidelines Use the **ip urlfilter urlf-server-log** command to enable Cisco IOS to send a log request immediately after the URL lookup request. The firewall will not make a URL lookup request if the destination IP address is in the cache, but it will still make a log request to the server. (The log request contains the URL, host name, source IP address, and the destination IP address.) The server records the log request into its own log server so your can view this information as necessary.

Examples The following example shows how to enable system message logging on the URL filter server:

```
ip urlfilter urlf-server-log
```

ip verify unicast reverse-path



Note

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

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

ip verify unicast reverse-path [*list*]

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

Syntax Description	<i>list</i>
	(Optional) Specifies a numbered access control list (ACL) in the following ranges: <ul style="list-style-type: none"> • 1 to 99 (IP standard access list) • 100 to 199 (IP extended access list) • 1300 to 1999 (IP standard access list, expanded range) • 2000 to 2699 (IP extended access list, expanded range)

Defaults Unicast RPF is disabled.

Command Modes Interface configuration mode

Command History	Release	Modification
	11.1(CC), 12.0	This command was introduced. This command was not included in Cisco IOS Release 11.2 or 11.3
	12.1(2)T	Added ACL support using the <i>list</i> argument. Added per-interface statistics on dropped or suppressed packets.
	12.0(15) S	The ip verify unicast source reachable-via command replaced this command, and the following keywords were added: allow-default , allow-self-ping , rx , and any .
	12.1(8a)E	The ip verify unicast source reachable-via command was integrated into Cisco IOS Release 12.1(8a)E.
	12.2(13)T	The ip verify unicast source reachable-via command was integrated into Cisco IOS Release 12.2(13)T.
	12.2(14)S	The ip verify unicast source reachable-via command was integrated into Cisco IOS Release 12.2(14)S.

Usage Guidelines

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

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

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

**Note**

It is very important for CEF to be configured globally in the router. Unicast RPF will not work without CEF.

**Note**

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

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

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

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

Where to Use RPF in Your Network

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

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

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

Examples

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

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

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

```
ip cef distributed
!
interface Serial 5/0/0
 description Connection to Upstream ISP
 ip address 192.168.200.225 255.255.255.255
 no ip redirects
 no ip directed-broadcast
 no ip proxy-arp
 ip verify unicast reverse-path
 ip access-group 111 in
 ip access-group 110 out
!
access-list 110 permit ip 192.168.202.128 10.0.0.31 any
access-list 110 deny ip any any log
access-list 111 deny ip host 10.0.0.0 any log
access-list 111 deny ip 172.16.0.0 255.255.255.255 any log
access-list 111 deny ip 10.0.0.0 255.255.255.255 any log
access-list 111 deny ip 172.16.0.0 255.255.255.255 any log
access-list 111 deny ip 192.168.0.0 255.255.255.255 any log
access-list 111 deny ip 209.165.202.129 10.0.0.31 any log
access-list 111 permit ip any any
```

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

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

```
ip cef distributed
!
int eth0/1/1
 ip address 192.168.200.1 255.255.255.255
```

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

Related Commands


Command	Description
ip cef	Enables CEF on the route processor card.

ip verify unicast source reachable-via

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

ip verify unicast source reachable-via { **rx** | **any** } [**allow-default**] [**allow-self-ping**] [*list*]

no ip verify unicast source reachable-via

Syntax Description		
rx	Examines incoming packets to determine whether the source address is in the Forwarding Information Base (FIB) and permits the packet only if the source is reachable through the interface on which the packet was received (sometimes referred to as strict mode).	
any	Examines incoming packets to determine whether the source address is in the FIB and permits the packet if the source is reachable through any interface (sometimes referred to as loose mode).	
allow-default	(Optional) Allows the use of the default route for RPF verification.	
allow-self-ping	(Optional) Allows a router to ping its own interface or interfaces.	
		
	Caution	Use caution when enabling the allow-self-ping keyword. This keyword opens a denial-of-service (DoS) hole.
<i>list</i>	(Optional) Specifies a numbered access control list (ACL) in the following ranges:	<ul style="list-style-type: none"> • 1 to 99 (IP standard access list) • 100 to 199 (IP extended access list) • 1300 to 2699 (IP standard access list, expanded range)

Command Default Unicast RPF is disabled.

Command Modes Interface configuration

Command History	Release	Modification
	11.1(CC), 12.0	This command was introduced. This command was not included in Cisco IOS Release 11.2 or 11.3.
	12.1(2)T	Added ACL support using the <i>list</i> argument. Added per-interface statistics on dropped or suppressed packets.
	12.0(15) S	This command replaced the ip verify unicast reverse-path command, and the following keywords were added: allow-default , allow-self-ping , rx , and any .
	12.1(8a)E	This command was integrated into Cisco IOS Release 12.1(8a)E.

Release	Modification
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use the **ip verify unicast source reachable-via** interface command to mitigate problems caused by malformed or forged (spoofed) IP source addresses that pass through a router. Malformed or forged source addresses can indicate DoS attacks on the basis of source IP address spoofing.

When Unicast RPF is enabled on an interface, the router examines all packets that are received on that interface. The router checks to make sure that the source address appears in the FIB. If the **rx** keyword is selected, the source address must match the interface on which the packet was received. If the **any** keyword is selected, the source address must be present only in the FIB. This ability to “look backwards” is available only when Cisco Express Forwarding (CEF) is enabled on the router because the lookup relies on the presence of the FIB. CEF generates the FIB as part of its operation.



Note

If the source address of an incoming packet is resolved to a null adjacency, the packet will be dropped. The null interface is treated as an invalid interface by the new form of the Unicast RPF command. The older form of the command syntax did not exhibit this behavior.

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



Note

It is very important for CEF to be configured globally in the router. Unicast RPF will not work without CEF.



Note

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

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

If no ACL is specified in the **ip verify unicast source reachable-via** command, the router drops the forged or malformed packet immediately and no ACL logging occurs. The router and interface Unicast RPF counters are updated.

Unicast RPF events can be logged by specifying the logging option for the ACL entries that are used by the **ip verify unicast source reachable-via** command. Log information can be used to gather such information about the attack, as source address, time, and so on.

Strict Mode RPF

If the source address is in the FIB and reachable only through the interface on which the packet was received, the packet is passed. The syntax for this method is **ip verify unicast source reachable-via rx**.

Exists-Only (or Loose Mode) RPF

If the source address is in the FIB and reachable through any interface on the router, the packet is passed. The syntax for this method is **ip verify unicast source reachable-via any**.

Because this Unicast RPF option passes packets regardless of which interface the packet enters, it is often used on Internet Service Provider (ISP) routers that are “peered” with other ISP routers (where asymmetrical routing typically occurs). Packets using source addresses that have not been allocated on the Internet, which are often used for spoofed source addresses, are dropped by this Unicast RPF option. All other packets that have an entry in the FIB are passed.

allow-default

Normally, sources found in the FIB, but only by way of the default route will be dropped. Specifying the **allow-default** keyword option will override this behavior. You must specify the **allow-default** keyword in the command to permit Unicast RPF to successfully match on prefixes that are known through the default route to pass these packets.

allow-self-ping

This keyword allows the router to ping its own interface or interfaces. By default, when Unicast RPF is enabled, packets that are generated by the router and destined to the router are dropped, thereby, making certain troubleshooting and management tasks difficult to accomplish. Issue the **allow-self-ping** keyword to enable self-pinging.



Caution

Caution should be used when enabling the **allow-self-ping** keyword because this option opens a potential DoS hole.

Where to Use RPF in Your Network

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



Note

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

Unicast RPF loose mode may be used on interfaces in which asymmetric paths allow packets from valid source networks (networks contained in the FIB). Routers that are in the core of the ISP network have no guarantee that the best forwarding path out of the router will be the path selected for packets returning to the router.

Examples

The following example uses a very simple single-homed ISP connection to demonstrate the concept of Unicast RPF. In this example, an ISP peering router is connected via a single serial interface to one upstream ISP. Hence, traffic flows into and out of the ISP will be symmetric. Because traffic flows will be symmetric, a Unicast RPF strict-mode deployment can be configured.

```
ip cef
! or "ip cef distributed" for Route Switch Processor+Versatile Interface Processor-
(RSP+VIP-) based routers.
!
interface Serial5/0/0
description - link to upstream ISP (single-homed)
ip address 192.168.200.225 255.255.255.252
no ip redirects
no ip directed-broadcasts
no ip proxy-arp
ip verify unicast source reachable-via
```

Related Commands

Command	Description
ip cef	Enables CEF on the route processor card.

ip vrf forwarding (server-group)

To configure the Virtual Private Network (VPN) routing and forwarding (VRF) reference of an authentication, authorization, and accounting (AAA) RADIUS server group, use the **ip vrf forwarding** command in server-group configuration mode. To enable server groups to use the global (default) routing table, use the **no** form of this command.

ip vrf forwarding *vrf-name*

no ip vrf forwarding *vrf-name*

Syntax Description	<i>vrf-name</i>	Name assigned to a VRF.
---------------------------	-----------------	-------------------------

Defaults Server groups use the global routing table.

Command Modes Server-group configuration

Command History	Release	Modification
	12.2(2)DD	This command was introduced on the Cisco 7200 series and Cisco 7401ASR.
	12.2(4)B	This command was integrated into Cisco IOS Release 12.2(4)B.
	12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.

Usage Guidelines Use the **ip vrf forwarding** command to specify a VRF for a AAA RADIUS server group. This command enables dial users to utilize AAA servers in different routing domains.

Examples The following example shows how to configure the VRF user to reference the RADIUS server in a different VRF server group:

```
aaa group server radius sg_global
  server-private 172.16.0.0 timeout 5 retransmit 3
!
aaa group server radius sg_water
  server-private 10.10.0.0 timeout 5 retransmit 3 key water
  ip vrf forwarding water
```

Related Commands	Command	Description
	aaa group server radius	Groups different RADIUS server hosts into distinct lists and distinct methods.
	server-private	Configures the IP address of the private RADIUS server for the group server.

ip-address (ca-trustpoint)

To specify a dotted IP address or an interface that will be included in the certificate request, use the **ip-address** command in ca-trustpoint configuration mode. To restore the default behavior, use the **no** form of this command.

ip-address { *ip-address* | *interface* }

no ip-address

Syntax Description		
	<i>ip-address</i>	Specifies a dotted IP address that will be included in the certificate request.
	<i>interface</i>	Specifies an interface, from which the router can get an IP address, that will be included in the certificate request.

Defaults You are prompted for the IP address during certificate enrollment.

Command Modes Ca-trustpoint configuration

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

Usage Guidelines

Before you can issue this command, you must enable the **crypto ca trustpoint** command, which declares the certification authority (CA) that your router should use and enters ca-trustpoint configuration mode. The **ip address** command is a subcommand that allows you to specify a certificate enrollment parameter.

Use the **ip-address** command to include the IP address of the specified interface in the certificate request or to specify that an IP address should not be included in the certificate request.

If this command is enabled, you will not be prompted for an IP address during certificate enrollment.

Examples The following example shows how to include the IP address of the Ethernet-0 interface in the certificate request for the trustpoint "frog":

```
crypto ca trustpoint frog
 enrollment url http://frog.phoobin.com/
 subject-name OU=Spiral Dept., O=tiedye.com
 ip-address ethernet-0
```

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

isakmp authorization list

To configure an Internet Key Exchange (IKE) shared secret using the authentication, authorization, and accounting (AAA) server in an Internet Security Association and Key Management Protocol (ISAKMP) profile, use the **isakmp authorization list** command in ISAKMP profile configuration mode. To disable the shared secret, use the **no** form of this command.

isakmp authorization list *list-name*

no isakmp authorization list *list-name*

Syntax Description	<i>list-name</i>	AAA authorization list used for configuration mode attributes or preshared keys for aggressive mode.
---------------------------	------------------	--

Defaults	No default behaviors or values
-----------------	--------------------------------

Command Modes	ISAKMP profile configuration
----------------------	------------------------------

Command History	<table border="1"> <thead> <tr> <th style="border-right: none;">Release</th> <th style="border-left: none;">Modification</th> </tr> </thead> <tbody> <tr> <td style="border-right: none;">12.2(15)T</td> <td style="border-left: none;">This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	12.2(15)T	This command was introduced.
Release	Modification				
12.2(15)T	This command was introduced.				

Usage Guidelines	This command allows you to retrieve a shared secret from an AAA server.
-------------------------	---

Examples	<p>The following example shows that an IKE shared secret is configured using an AAA server on a router:</p> <pre>crypto isakmp profile vpnprofile isakmp authorization list ikessaaalist</pre>
-----------------	---

Related Commands	<table border="1"> <thead> <tr> <th style="border-right: none;">Command</th> <th style="border-left: none;">Description</th> </tr> </thead> <tbody> <tr> <td style="border-right: none;">aaa authorization</td> <td style="border-left: none;">Sets parameters that restrict user access to a network.</td> </tr> </tbody> </table>	Command	Description	aaa authorization	Sets parameters that restrict user access to a network.
Command	Description				
aaa authorization	Sets parameters that restrict user access to a network.				

keepalive (isakmp profile)

To allow the gateway to send dead peer detection (DPD) messages to the peer, use the **keepalive** command in Internet Security Association Key Management Protocol (ISAKMP) profile configuration mode. To return to the default, use the **no** form of this command.

keepalive *seconds* **retry** *retry-seconds*

no keepalive *seconds* **retry** *retry-seconds*

Syntax Description		
<i>seconds</i>		Number of seconds between DPD messages. The range is from 10 to 3600 seconds.
retry <i>retry-seconds</i>		Number of seconds between retries if DPD message fails. The range is from 2 to 60 seconds.

Defaults If this command is not configured, a DPD message is not sent to the client.

Command Modes ISAKMP profile configuration

Command History	Release	Modification
	12.2(15)T	This command was introduced.

Usage Guidelines Use this command to enable the gateway (instead of the client) to send DPD messages to the client. Internet Key Exchange (IKE) DPD is a new keepalive scheme that sends messages to let the router know that the client is still connected.

Examples The following example shows that DPD messages have been configured to be sent every 60 seconds and every 5 seconds between retries if the peer does not respond:

```
crypto isakmp profile vpnprofile
  keepalive 60 retry 5
```

kerberos clients mandatory

To cause the **rsh**, **rnp**, **rlogin**, and **telnet** commands to fail if they cannot negotiate the Kerberos protocol with the remote server, use the **kerberos clients mandatory** command in global configuration mode. To make Kerberos optional, use the **no** form of this command.

kerberos clients mandatory

no kerberos clients mandatory

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	11.2	This command was introduced.

Usage Guidelines If this command is not configured and the user has Kerberos credentials stored locally, the **rsh**, **rnp**, **rlogin**, and **telnet** commands attempt to negotiate the Kerberos protocol with the remote server and will use the non-Kerberized protocols if unsuccessful.

If this command is not configured and the user has no Kerberos credentials, the standard protocols for **rnp** and **rsh** are used to negotiate.

Examples The following example causes the **rsh**, **rnp**, **rlogin**, and **telnet** commands to fail if they cannot negotiate the Kerberos protocol with the remote server:

```
kerberos clients mandatory
```

Related Commands	Command	Description
	connect	Logs in to a host that supports Telnet, rlogin, or LAT.
	kerberos credentials forward	Forces all network application clients on the router to forward the Kerberos credentials of users upon successful Kerberos authentication.
	rlogin	Logs in to a UNIX host using rlogin.
	rsh	Executes a command remotely on a remote rsh host.
	telnet	Logs in to a host that supports Telnet.

kerberos credentials forward

To force all network application clients on the router to forward users' Kerberos credentials upon successful Kerberos authentication, use the **kerberos credentials forward** command in global configuration mode. To turn off forwarding of Kerberos credentials, use the **no** form of this command.

kerberos credentials forward

no kerberos credentials forward

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Global configuration

Release	Modification
11.2	This command was introduced.

Usage Guidelines Enable credentials forwarding to have users' ticket granting tickets (TGTs) forwarded to the host on which they authenticate. In this way, users can connect to multiple hosts in the Kerberos realm without running the KINIT program each time they need to get a TGT.

Examples The following example forces all network application clients on the router to forward users' Kerberos credentials upon successful Kerberos authentication:

```
kerberos credentials forward
```

Command	Description
connect	Logs in to a host that supports Telnet, rlogin, or LAT.
rlogin	Logs in to a UNIX host using rlogin.
rsh	Executes a command remotely on a remote rsh host.
telnet	Logs in to a host that supports Telnet.

kerberos instance map

To map Kerberos instances to Cisco IOS privilege levels, use the **kerberos instance map** command in global configuration mode. To remove a Kerberos instance map, use the **no** form of this command.

kerberos instance map *instance privilege-level*

no kerberos instance map *instance*

Syntax Description	Parameter	Description
	<i>instance</i>	Name of a Kerberos instance.
	<i>privilege-level</i>	The privilege level at which a user is set if the user's Kerberos principal contains the matching Kerberos instance. You can specify up to 16 privilege levels, using numbers 0 through 15. Level 1 is normal EXEC-mode user privileges.

Defaults Privilege level 1

Command Modes Global configuration

Command History	Release	Modification
	11.2	This command was introduced.

Usage Guidelines Use this command to create user instances with access to administrative commands.

Examples The following example sets the privilege level to 15 for authenticated Kerberos users with the *admin* instance in Kerberos realm:

```
kerberos instance map admin 15
```

Related Commands	Command	Description
	aaa authorization	Sets parameters that restrict user access to a network.

kerberos local-realm

To specify the Kerberos realm in which the router is located, use the **kerberos local-realm** command in global configuration mode. To remove the specified Kerberos realm from this router, use the **no** form of this command.

kerberos local-realm *kerberos-realm*

no kerberos local-realm

Syntax Description	<i>kerberos-realm</i>	The name of the default Kerberos realm. A Kerberos realm consists of users, hosts, and network services that are registered to a Kerberos server. The Kerberos realm must be in uppercase characters.
Defaults	Disabled	
Command Modes	Global configuration	
Command History	Release	Modification
	11.1	This command was introduced.
Usage Guidelines	The router can be located in more than one realm at a time. However, there can only be one instance of Kerberos local-realm. The realm specified with this command is the default realm.	
Examples	The following example specify the Kerberos realm in which the router is located as EXAMPLE.COM: kerberos local-realm EXAMPLE.COM	
Related Commands	Command	Description
	kerberos preauth	Specifies a preauthentication method to use to communicate with the KDC.
	kerberos realm	Maps a host name or DNS domain to a Kerberos realm.
	kerberos server	Specifies the location of the Kerberos server for a given Kerberos realm.
	kerberos srvtab entry	Specifies a krb5 SRVTAB entry.
	kerberos srvtab remote	Retrieves a SRVTAB file from a remote host and automatically generate a Kerberos SRVTAB entry configuration.

kerberos preauth

To specify a preauthentication method to use to communicate with the key distribution center (KDC), use the **kerberos preauth** command in global configuration mode. To disable Kerberos preauthentication, use the **no** form of this command.

kerberos preauth [**encrypted-unix-timestamp** | **encrypted-kerberos-timestamp** | **none**]

no kerberos preauth

Syntax Description	
encrypted-unix-timestamp	(Optional) Use an encrypted UNIX timestamp as a quick authentication method when communicating with the KDC.
encrypted-kerberos-timestamp	(Optional) Use the RFC1510 kerberos timestamp as a quick authentication method when communicating with the KDC.
none	(Optional) Do not use Kerberos preauthentication.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	11.2	This command was introduced.

Usage Guidelines It is more secure to use a preauthentication for communications with the KDC. However, communication with the KDC will fail if the KDC does not support this particular version of **kerberos preauth**. If that happens, turn off the preauthentication with the **none** option.

The **no** form of this command is equivalent to using the **none** keyword.

Examples The following example enables Kerberos preauthentication:

```
kerberos preauth encrypted-unix-timestamp
```

The following example disables Kerberos preauthentication:

```
kerberos preauth none
```

Related Commands	Command	Description
	kerberos local-realm	Specifies the Kerberos realm in which the router is located.
	kerberos server	Specifies the location of the Kerberos server for a given Kerberos realm.

Command	Description
kerberos srvtab entry	Specifies a krb5 SRVTAB entry.
kerberos srvtab remote	Retrieves a SRVTAB file from a remote host and automatically generate a Kerberos SRVTAB entry configuration.

kerberos realm

To map a host name or Domain Name System (DNS) domain to a Kerberos realm, use the **kerberos realm** command in global configuration mode. To remove a Kerberos realm map, use the **no** form of this command.

```
kerberos realm {dns-domain | host} kerberos-realm
```

```
no kerberos realm {dns-domain | host} kerberos-realm
```

Syntax Description		
	<i>dns-domain</i>	Name of a DNS domain or host.
	<i>host</i>	Name of a DNS host.
	<i>kerberos-realm</i>	Name of the Kerberos realm to which the specified domain or host belongs.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	11.1	This command was introduced.

Usage Guidelines DNS domains are specified with a leading dot (.) character; host names cannot begin with a dot (.) character. There can be multiple entries of this line.

A Kerberos realm consists of users, hosts, and network services that are registered to a Kerberos server. The Kerberos realm must be in uppercase characters. The router can be located in more than one realm at a time. Kerberos realm names must be in all uppercase characters.

Examples The following example maps the domain name “example.com” to the Kerberos realm, EXAMPLE.COM:

```
kerberos realm .example.com EXAMPLE.COM
```

Related Commands	Command	Description
	kerberos local-realm	Specifies the Kerberos realm in which the router is located.
	kerberos server	Specifies the location of the Kerberos server for a given Kerberos realm.
	kerberos srvtab entry	Specifies a krb5 SRVTAB entry.
	kerberos srvtab remote	Retrieves a SRVTAB file from a remote host and automatically generates a Kerberos SRVTAB entry configuration.

kerberos server

To specify the location of the Kerberos server for a given Kerberos realm, use the **kerberos server** command in global configuration mode. To remove a Kerberos server for a specified Kerberos realm, use the **no** form of this command.

kerberos server *kerberos-realm* {*host-name* | *ip-address*} [*port-number*]

no kerberos server *kerberos-realm* {*host-name* | *ip-address*}

Syntax Description		
<i>kerberos-realm</i>	Name of the Kerberos realm. A Kerberos realm consists of users, hosts, and network services that are registered to a Kerberos server. The Kerberos realm must be in uppercase letters.	
<i>host-name</i>	Name of the host functioning as a Kerberos server for the specified Kerberos realm (translated into an IP address at the time of entry).	
<i>ip-address</i>	IP address of the host functioning as the Kerberos server for the specified Kerberos realm.	
<i>port-number</i>	(Optional) Port that the key distribution center (KDC) monitors (defaults to 88).	

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	11.1	This command was introduced.

Usage Guidelines Use the **kerberos server** command to specify the location of the Kerberos server for a given realm.

Examples The following example specifies 192.168.47.66 as the Kerberos server for the Kerberos realm EXAMPLE.COM:

```
kerberos server EXAMPLE.COM 192.168.47.66
```

Related Commands	Command	Description
	kerberos local-realm	Specifies the Kerberos realm in which the router is located.
	kerberos realm	Maps a host name or DNS domain to a Kerberos realm.
	kerberos srvtab entry	Specifies a krb5 SRVTAB entry.
	kerberos srvtab remote	Retrieves a SRVTAB file from a remote host and automatically generates a Kerberos SRVTAB entry configuration.

kerberos srvtab entry

To retrieve a SRVTAB file from a remote host and automatically generate a Kerberos SRVTAB entry configuration, use the **kerberos srvtab entry** command in global configuration mode. To remove a SRVTAB entry from the router's configuration, use the **no** form of this command.

kerberos srvtab entry *kerberos-principal principal-type timestamp key-version number key-type key-length encrypted-keytab*

no kerberos srvtab entry *kerberos-principal principal-type*

Syntax Description

<i>kerberos-principal</i>	A service on the router.
<i>principal-type</i>	Version of the Kerberos SRVTAB.
<i>timestamp</i>	Number representing the date and time the SRVTAB entry was created.
<i>key-version number</i>	Version of the encryption key format.
<i>key-type</i>	Type of encryption used.
<i>key-length</i>	Length, in bytes, of the encryption key.
<i>encrypted-keytab</i>	Secret key the router shares with the key distribution center (KDC). It is encrypted with the private Data Encryption Standard (DES) key (if available) when you write out your configuration.

Defaults

No default behavior or values.

Command Modes

Global configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

When you use the **kerberos srvtab remote** command to copy the SRVTAB file from a remote host (generally the KDC), it parses the information in this file and stores it in the router's running configuration in the **kerberos srvtab entry** format. The key for each SRVTAB entry is encrypted with a private DES key if one is defined on the router. To ensure that the SRVTAB is available (that is, that it does not need to be acquired from the KDC) when you reboot the router, use the **write memory** router configuration command to write the router's running configuration to NVRAM.

If you reload a configuration, with a SRVTAB encrypted with a private DES key, on to a router that does not have a private DES key defined, the router displays a message informing you that the SRVTAB entry has been corrupted, and discards the entry.

If you change the private DES key and reload an old version of the router's configuration that contains SRVTAB entries encrypted with the old private DES keys, the router will restore your Kerberos SRVTAB entries, but the SRVTAB keys will be corrupted. In this case, you must delete your old Kerberos SRVTAB entries and reload your Kerberos SRVTABs on to the router using the **kerberos srvtab remote** command.

Although you can configure **kerberos srvtab entry** on the router manually, generally you would not do this because the keytab is encrypted automatically by the router when you copy the SRVTAB using the **kerberos srvtab remote** command.

Examples

In the following example, `host/new-router.example.com@EXAMPLE.COM` is the host, 0 is the type, 817680774 is the timestamp, 1 is the version of the key, 1 indicates the DES is the encryption type, 8 is the number of bytes, and `.cCN.YoU.okK` is the encrypted key:

```
kerberos srvtab entry host/new-router.example.com@EXAMPLE.COM 0 817680774 1 1 8
.cCN.YoU.okK
```

Related Commands

Command	Description
kerberos srvtab remote	Retrieves a krb5 SRVTAB file from the specified host.
key config-key	Defines a private DES key for the router.

kerberos srvtab remote

To retrieve a SRVTAB file from a remote host and automatically generate a Kerberos SRVTAB entry configuration, use the **kerberos srvtab remote** command in global configuration mode.

```
kerberos srvtab remote {boot_device:URL}
```

Syntax Description	<i>URL</i>	Machine that has the Kerberos SRVTAB file.
	<i>ip-address</i>	IP address of the machine that has the Kerberos SRVTAB file.
	<i>filename</i>	Name of the SRVTAB file.

Defaults No default behavior or values.

Command Modes Global configuration

Command History	Release	Modification
	11.2	This command was introduced.

Usage Guidelines When you use the **kerberos srvtab remote** command to copy the SRVTAB file from the remote host (generally the key distribution center [KDC]), it parses the information in this file and stores it in the router's running configuration in the **kerberos srvtab entry** format. The key for each SRVTAB entry is encrypted with the private Data Encryption Standard (DES) key if one is defined on the router. To ensure that the SRVTAB is available (that is, that it does not need to be acquired from the KDC) when you reboot the router, use the **write memory** configuration command to write the router's running configuration to NVRAM.

Examples The following example copies the SRVTAB file residing on b1.example.com to a router named s1.example.com:

```
kerberos srvtab remote tftp://b1.example.com/s1.example.com-new-srvtab
```

Related Commands	Command	Description
	kerberos srvtab entry	Retrieves a SRVTAB file from a remote host and automatically generate a Kerberos SRVTAB entry configuration.
	key config-key	Defines a private DES key for the router.

key (isakmp-group)

To specify the Internet Key Exchange (IKE) preshared key for group policy attribute definition, use the **key** command in Internet Security Association Key Management Protocol (ISAKMP) group configuration mode. To remove a preshared key, use the **no** form of this command.

key *name*

no key *name*

Syntax Description	<i>name</i>	IKE preshared key that matches the password entered on the client.
	Note	This value must match the “password” field that is defined in the Cisco VPN Client 3.x configuration GUI.

Defaults No default behavior or values.

Command Modes ISAKMP group configuration

Command History	Release	Modification
		12.2(8)T

Usage Guidelines Use the **key** command to specify the IKE preshared key when defining group policy information for Mode Configuration push. (This command follows the **crypto isakmp client configuration group** command.) You *must* configure this command if the client identifies itself to the router with a preshared key. (You do not have to enable this command if the client uses a certificate for identification.)

Examples The following example shows how to specify the preshared key “cisco”:

```
crypto isakmp client configuration group default
key cisco
dns 2.2.2.2 2.3.2.3
pool dog
acl 199
```

Related Commands	Command	Description
		crypto isakmp client configuration group

key config-key

To define a private DES key for the router, use the **key config-key** command in global configuration mode. To delete a private Data Encryption Standard (DES) key from the router, use the **no** form of this command.

key config-key 1 string

no key config-key 1 string

Syntax Description	1	Key number. This number is always 1.
	<i>string</i>	Private DES key (can be up to eight alphanumeric characters).

Defaults No DES-key defined.

Command Modes Global configuration

Command History	Release	Modification
		11.2

Usage Guidelines This command defines a private DES key for the router that will not show up in the router configuration. This private DES key can be used to DES-encrypt certain parts of the router's configuration.



Caution

The private DES key is unrecoverable. If you encrypt part of your configuration with the private DES key and lose or forget the key, you will not be able to recover the encrypted data.

Examples The following example sets *keyxx* as the private DES key on the router:

```
key config-key 1 keyxx
```

Related Commands	Command	Description
		kerberos srvtab entry
	kerberos srvtab remote	Retrieves a SRVTAB file from a remote host and automatically generates a Kerberos SRVTAB entry configuration.

keyring

To configure a keyring with an Internet Security Association and Key Management Protocol (ISAKMP) profile, use the **keyring** command in ISAKMP profile configuration mode. To remove the keyring from the ISAKMP profile, use the **no** form of this command.

keyring *keyring-name*

no keyring *keyring-name*

Syntax Description	<i>keyring-name</i>	The keyring name, which must match the keyring name that was defined in the global configuration.
---------------------------	---------------------	---

Defaults If this command is not used, the ISAKMP profile uses the keys defined in the global configuration.

Command Modes ISAKMP profile configuration

Command History	Release	Modification
	12.2(15)T	This command was introduced.

Usage Guidelines The ISAKMP profile successfully completes authentication of peers if the peer keys are defined in the keyring that is attached to this profile. If no keyring is defined in the profile, the global keys that were defined in the global configuration are used.

Examples The following example shows that “vpnkeyring” is configured as the keyring name:

```
crypto isakmp profile vpnprofile
  keyring vpnkeyring
```

key-string (IKE)

To specify the Rivest, Shamir, and Adelman (RSA) public key of the remote peer, use the **key-string** command in public key configuration mode. To remove the RSA public key, use the **no** form of this command.

key-string *key-string*

no key-string *key-string*

Syntax Description	<i>key-string</i>	Enter the key in hexadecimal format. While entering the key data, you can press Return to continue entering data.
---------------------------	-------------------	---

Defaults	No default behavior or values
-----------------	-------------------------------

Command Modes	Public key configuration
----------------------	--------------------------

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines	<p>Before using this command, you must enter the rsa-pubkey command in the crypto keyring mode. If possible, to avoid mistakes, you should cut and paste the key data (instead of attempting to type in the data).</p> <p>To complete the command, you must return to the global configuration mode by typing quit at the config-pubkey prompt.</p>
-------------------------	---

Examples	<p>The following example manually specifies the RSA public keys of an IP Security (IPSec) peer:</p>
-----------------	---

```
Router(config)# crypto keyring vpnkeyring
Router(conf-keyring)# rsa-pubkey name host.vpn.com
Router(config-pubkey-key)# address 10.5.5.1
Router(config-pubkey)# key-string
Router(config-pubkey)# 00302017 4A7D385B 1234EF29 335FC973
Router(config-pubkey)# 2DD50A37 C4F4B0FD 9DADE748 429618D5
Router(config-pubkey)# 18242BA3 2EDFBDD3 4296142A DDF7D3D8
Router(config-pubkey)# 08407685 2F2190A0 0B43F1BD 9A8A26DB
Router(config-pubkey)# 07953829 791FCDE9 A98420F0 6A82045B
Router(config-pubkey)# 90288A26 DBC64468 7789F76E EE21
Router(config-pubkey)# quit
Router(config-pubkey-key)# exit
Router(conf-keyring)# exit
```

Related Commands

Command	Description
crypto keyring	Defines a crypto keyring.
rsa-pubkey	Defines the RSA public key to be used for encryption or signatures during IKE authentication.
show crypto keyring	Displays keyrings on your router.

lifetime (IKE policy)

To specify the lifetime of an Internet Key Exchange (IKE) security association (SA), use the **lifetime** command in Internet Security Association Key Management Protocol (ISAKMP) policy configuration mode. To reset the SA lifetime to the default value, use the **no** form of this command.

lifetime *seconds*

no lifetime

Syntax Description	<i>seconds</i>	Number of many seconds for each each SA should exist before expiring. Use an integer from 60 to 86,400 seconds, which is the default value.
---------------------------	----------------	---

Defaults	86,400 seconds (one day)
-----------------	--------------------------

Command Modes	ISAKMP policy configuration
----------------------	-----------------------------

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines	Use this command to specify how long an IKE SA exists before expiring.
-------------------------	--

When IKE begins negotiations, the first thing it does is agree upon the security parameters for its own session. The agreed-upon parameters are then referenced by an SA at each peer. The SA is retained by each peer until the SA's lifetime expires. Before an SA expires, it can be reused by subsequent IKE negotiations, which can save time when setting up new IPsec SAs. Before an SA expires, it can be reused by subsequent IKE negotiations, which can save time when setting up new IPsec SAs. New IPsec SAs are negotiated before current IPsec SAs expire.

So, to save setup time for IPsec, configure a longer IKE SA lifetime. However, shorter lifetimes limit the exposure to attackers of this SA. The longer an SA is used, the more encrypted traffic can be gathered by an attacker and possibly used in an attack.

Note that when your local peer initiates an IKE negotiation between itself and a remote peer, an IKE policy can be selected only if the lifetime of the remote peer's policy is shorter than or equal to the lifetime of the local peer's policy. Then, if the lifetimes are not equal, the shorter lifetime will be selected. To restate this behavior: If the two peer's policies' lifetimes are not the same, the initiating peer's lifetime must be longer and the responding peer's lifetime must be shorter, and the shorter lifetime will be used.

Examples	The following example configures an IKE policy with a security association lifetime of 600 seconds (10 minutes), and all other parameters are set to the defaults:
-----------------	--

```
crypto isakmp policy 15
  lifetime 600
exit
```

Related Commands

Command	Description
authentication (IKE policy)	Specifies the authentication method within an IKE policy.
crypto isakmp policy	Defines an IKE policy.
encryption (IKE policy)	Specifies the encryption algorithm within an IKE policy.
group (IKE policy)	Specifies the Diffie-Hellman group identifier within an IKE policy.
hash (IKE policy)	Specifies the hash algorithm within an IKE policy.
show crypto isakmp policy	Displays the parameters for each IKE policy.

login authentication

To enable authentication, authorization, and accounting (AAA) authentication for logins, use the **login authentication** command in line configuration mode. To return to the default specified by the **aaa authentication login** command, use the **no** form of this command.

login authentication { **default** | *list-name* }

no login authentication { **default** | *list-name* }

Syntax Description

default	Uses the default list created with the aaa authentication login command.
<i>list-name</i>	Uses the indicated list created with the aaa authentication login command.

Defaults

Uses the default set with **aaa authentication login**.

Command Modes

Line configuration

Command History

Release	Modification
10.3	This command was introduced.

Usage Guidelines

This command is a per-line command used with AAA that specifies the name of a list of AAA authentication methods to try at login. If no list is specified, the default list is used (whether or not it is specified in the command line).



Caution

If you use a *list-name* value that was not configured with the **aaa authentication login** command, you will disable login on this line.

Entering the **no** version of **login authentication** has the same effect as entering the command with the **default** keyword.

Before issuing this command, create a list of authentication processes by using the global configuration **aaa authentication login** command.

Examples

The following example specifies that the default AAA authentication is to be used on line 4:

```
line 4
 login authentication default
```

The following example specifies that the AAA authentication list called *list1* is to be used on line 7:

```
line 7
 login authentication list1
```

Related Commands

Command	Description
aaa authentication login	Sets AAA authentication at login.

match address (IPSec)

To specify an extended access list for a crypto map entry, use the **match address** command in crypto map configuration mode. To remove the extended access list from a crypto map entry, use the **no** form of this command.

match address [*access-list-id* | *name*]

no match address [*access-list-id* | *name*]

Syntax Description

<i>access-list-id</i>	(Optional) Identifies the extended access list by its name or number. This value should match the <i>access-list-number</i> or <i>name</i> argument of the extended access list being matched.
<i>name</i>	(Optional) Identifies the named encryption access list. This name should match the <i>name</i> argument of the named encryption access list being matched.

Defaults

No access lists are matched to the crypto map entry.

Command Modes

Crypto map configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

This command is required for all static crypto map entries. If you are defining a dynamic crypto map entry (with the **crypto dynamic-map** command), this command is not required but is strongly recommended.

Use this command to assign an extended access list to a crypto map entry. You also need to define this access list using the **access-list** or **ip access-list extended** commands.

The extended access list specified with this command will be used by IPSec to determine which traffic should be protected by crypto and which traffic does not need crypto protection. (Traffic that is permitted by the access list will be protected. Traffic that is denied by the access list will not be protected in the context of the corresponding crypto map entry.)

Note that the crypto access list is *not* used to determine whether to permit or deny traffic through the interface. An access list applied directly to the interface makes that determination.

The crypto access list specified by this command is used when evaluating both inbound and outbound traffic. Outbound traffic is evaluated against the crypto access lists specified by the interface's crypto map entries to determine if it should be protected by crypto and if so (if traffic matches a **permit** entry) which crypto policy applies. (If necessary, in the case of static IPSec crypto maps, new security associations are established using the data flow identity as specified in the **permit** entry; in the case of dynamic crypto map entries, if no SA exists, the packet is dropped.) After passing the regular access lists at the interface, inbound traffic is evaluated against the crypto access lists specified by the entries of the

interface's crypto map set to determine if it should be protected by crypto and, if so, which crypto policy applies. (In the case of IPSec, unprotected traffic is discarded because it should have been protected by IPSec.)

In the case of IPSec, the access list is also used to identify the flow for which the IPSec security associations are established. In the outbound case, the **permit** entry is used as the data flow identity (in general), while in the inbound case the data flow identity specified by the peer must be "permitted" by the crypto access list.

Examples

The following example shows the minimum required crypto map configuration when IKE will be used to establish the security associations. (This example is for a static crypto map.)

```
crypto map mymap 10 ipsec-isakmp
 match address 101
 set transform-set my_t_set1
 set peer 10.0.0.1
```

Related Commands

Command	Description
crypto dynamic-map	Creates a dynamic crypto map entry and enters the crypto map configuration command mode.
crypto map (global IPSec)	Creates or modifies a crypto map entry and enters the crypto map configuration mode.
crypto map (interface IPSec)	Applies a previously defined crypto map set to an interface.
crypto map local-address	Specifies and names an identifying interface to be used by the crypto map for IPSec traffic.
set peer (IPSec)	Specifies an IPSec peer in a crypto map entry.
set pfs	Specifies that IPSec should ask for perfect forward secrecy (PFS) when requesting new security associations for this crypto map entry, or that IPSec requires PFS when receiving requests for new security associations.
set security-association level per-host	Specifies that separate IPSec security associations should be requested for each source/destination host pair.
set security-association lifetime	Overrides (for a particular crypto map entry) the global lifetime value, which is used when negotiating IPSec security associations.
set session-key	Specifies the IPSec session keys within a crypto map entry.
set transform-set	Specifies which transform sets can be used with the crypto map entry.
show crypto map (IPSec)	Displays the crypto map configuration.

match certificate

To associate a certificate-based access control list (ACL) defined with the **crypto ca certificate map** command, use the **match certificate** command in ca-trustpoint configuration mode. To remove the association, use the **no** form of this command.

match certificate *certificate-map-label*

no match certificate *certificate-map-label*

Syntax Description

certificate-map-label Matches the *label* argument specified in a previously defined **crypto ca certificate map** command.

Defaults

No default match certificate is configured.

Command Modes

Ca-trustpoint configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.

Usage Guidelines

The **match certificate** subcommand associates the certificate-based ACL defined with the **crypto ca certificate map** command to the trustpoint. The *certificate-map-label* argument in the **match certificate** subcommand must match the *label* argument specified in a previously defined **crypto ca certificate map** command.

The certificate map with the label *certificate-map-label* must be defined before it can be used with the **match certificate** subcommand.

A certificate referenced in a **match certificate** subcommand may not be deleted until all references to the certificate map are removed from configured trustpoints (that is, no **match certificate** subcommands can reference the certificate map being deleted).

When the certificate of a peer has been verified, the certificate-based ACL as specified by the certificate map is checked. If the certificate of the peer matches the certificate ACL, or a certificate map is not associated with the trustpoint used to verify the certificate of the peer, the certificate of the peer is considered valid.

If the certificate map does not have any attributes defined, the certificate is rejected.

Examples

The following example shows a certificate-based ACL with the label “Group” defined in a **crypto ca certificate map** command and included in the **match certificate** subcommand of the **crypto ca trustpoint** command:

```
crypto ca certificate map Group 10
  subject-name co ou=WAN
  subject-name co o=Cisco
!
crypto ca trustpoint pki
  match certificate Group
```

Related Commands

Command	Description
crypto ca certificate map	Defines certificate-based ACLs.
crypto ca trustpoint	Declares the CA that your router should use.

match identity

To match an identity from a peer in an Internet Security Association and Key Management Protocol (ISAKMP) profile, use the **match identity** command in ISAKMP profile configuration mode. To remove the identity, use the **no** form of this command.

match identity { **group** *group-name* | **address** *address* [*mask*] [*fvr*] | **host** *host-name* / **host domain** *domain-name* | **user** *user-fqdn* / **user domain** *domain-name* }

no match identity { **group** *group-name* | **address** *address* [*mask*] [*fvr*] | **host** *host-name* / **host domain** *domain-name* | **user** *user-fqdn* / **user domain** *domain-name* }

Syntax Description

group <i>group-name</i>	A Unity group that matches identification (ID) type ID_KEY_ID. If Unity and main mode Rivest, Shamir, and Adelman (RSA) signatures are used, the <i>group-name</i> argument matches the Organizational Unit (OU) field of the Distinguished Name (DN).
address <i>address</i> [<i>mask</i>] [<i>fvr</i>]	An identity that matches the identity of type ID_IPV4_ADDR. <ul style="list-style-type: none"> <i>mask</i>—Use to match the range of the address. <i>fvr</i>—Use to match the address in the front door Virtual Route Forwarding (FVRF) Virtual Private Network (VPN) space.
host <i>host-name</i>	Identity that matches an identity of the type ID_FQDN.
host domain <i>domain-name</i>	Identity that matches an identity of the type ID_FQDN, whose fully qualified domain name (FQDN) ends with the domain name.
user <i>user-fqdn</i>	Identity that matches the FQDN.
user domain <i>domain-name</i>	Identity that matches the identities of the type ID_USER_FQDN. When the user domain keyword is present, all users having identities of the type ID_USER_FQDN and ending with “ <i>domain-name</i> ” will be matched.

Defaults

No default behavior or values

Command Modes

ISAKMP profile configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.

Usage Guidelines

There must be at least one **match identity** command in an ISAKMP profile configuration. The peers are mapped to an ISAKMP profile when their identities are matched (as given in the ID payload of the Internet Key Exchange [IKE] exchange) against the identities that are defined in the ISAKMP profile. To uniquely map to an ISAKMP profile, no two ISAKMP profiles should match the same identity. If the peer identity is matched in two ISAKMP profiles, the configuration is invalid.

Examples

The following example shows that the **match identity** command is configured:

```
crypto isakmp profile vpnprofile
  match identity group vpngroup
  match identity address 10.53.11.1
  match identity host domain vpn.com
  match identity host server.vpn.com
```

mode (IPSec)

To change the mode for a transform set, use the **mode** command in crypto transform configuration mode. To reset the mode to the default value of tunnel mode, use the **no** form of this command.

mode [tunnel | transport]

no mode

Syntax Description	tunnel / transport	(Optional) Specifies the mode for a transform set: either tunnel or transport mode. If neither tunnel nor transport is specified, the default (tunnel mode) is assigned.
---------------------------	-------------------------------	--

Defaults	Tunnel mode
-----------------	-------------

Command Modes	Crypto transform configuration
----------------------	--------------------------------

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines

Use this command to change the mode specified for the transform. This setting is only used when the traffic to be protected has the same IP addresses as the IPSec peers (this traffic can be encapsulated either in tunnel or transport mode). This setting is ignored for all other traffic (all other traffic is encapsulated in tunnel mode).

If the traffic to be protected has the same IP address as the IP Security peers and transport mode is specified, during negotiation the router will request transport mode but will accept either transport or tunnel mode. If tunnel mode is specified, the router will request tunnel mode and will accept only tunnel mode.

After you define a transform set, you are put into the crypto transform configuration mode. While in this mode you can change the mode to either tunnel or transport. This change applies only to the transform set just defined.

If you do not change the mode when you first define the transform set, but later decide you want to change the mode for the transform set, you must re-enter the transform set (specifying the transform name and all its transforms) and then change the mode.

If you use this command to change the mode, the change will only affect the negotiation of subsequent IPSec security associations via crypto map entries which specify this transform set. (If you want the new settings to take effect sooner, you can clear all or part of the security association database. See the **clear crypto sa** command for more details.)

Tunnel Mode

With tunnel mode, the entire original IP packet is protected (encrypted, authenticated, or both) and is encapsulated by the IPSec headers and trailers (an Encapsulation Security Protocol header and trailer, an Authentication Header, or both). Then a new IP header is prefixed to the packet, specifying the IPSec endpoints as the source and destination.

Tunnel mode can be used with any IP traffic. Tunnel mode must be used if IPsec is protecting traffic from hosts behind the IPsec peers. For example, tunnel mode is used with Virtual Private Networks (VPNs) where hosts on one protected network send packets to hosts on a different protected network via a pair of IPsec peers. With VPNs, the IPsec peers “tunnel” the protected traffic between the peers while the hosts on their protected networks are the session endpoints.

Transport Mode

With transport mode, only the payload (data) of the original IP packet is protected (encrypted, authenticated, or both). The payload is encapsulated by the IPsec headers and trailers (an ESP header and trailer, an AH header, or both). The original IP headers remain intact and are not protected by IPsec.

Use transport mode only when the IP traffic to be protected has IPsec peers as both the source and destination. For example, you could use transport mode to protect router management traffic. Specifying transport mode allows the router to negotiate with the remote peer whether to use transport or tunnel mode.

Examples

The following example defines a transform set and changes the mode to transport mode. The mode value only applies to IP traffic with the source and destination addresses at the local and remote IPsec peers.

```
crypto ipsec transform-set newer esp-des esp-sha-hmac
mode transport
exit
```

Related Commands

Command	Description
crypto ipsec transform-set	Defines a transform set—an acceptable combination of security protocols and algorithms.

named-key

To specify which peer's RSA public key you will manually configure and enter public key configuration mode, use the **named-key** command in public key chain configuration mode. This command should be used only when the router has a single interface that processes IP Security (IPSec).

named-key *key-name* [**encryption** | **signature**]

Syntax Description		
	<i>key-name</i>	Specifies the name of the remote peer's RSA keys. This is always the fully qualified domain name of the remote peer; for example, router.example.com.
	encryption	(Optional) Indicates that the RSA public key to be specified will be an encryption special-usage key.
	signature	(Optional) Indicates that the RSA public key to be specified will be a signature special-usage key.

Defaults If neither the **encryption** nor the **signature** keyword is used, general-purpose keys will be specified.

Command Modes Public key chain configuration.

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines Use this command or the **addressed-key** command to specify which IPSec peer's RSA public key you will manually configure next.

Follow this command with the **key-string** command to specify the key.

If you use the **named-key** command, you also need to use the **address** public key configuration command to specify the IP address of the peer.

If the IPSec remote peer generated general purpose RSA keys, do not use the **encryption** or **signature** keyword.

If the IPSec remote peer generated special usage keys, you must manually specify both keys: perform this command and the **key-string** command twice and use the **encryption** and **signature** keywords in turn.

Examples The following example manually specifies the RSA public keys of two IPSec peers. The peer at 10.5.5.1 uses general-purpose keys, and the other peer uses special-purpose keys.

```
crypto key pubkey-chain rsa
  named-key otherpeer.example.com
  address 10.5.5.1
  key-string
  005C300D 06092A86 4886F70D 01010105
  00034B00 30480241 00C5E23B 55D6AB22
```

```

04AEF1BA A54028A6 9ACC01C5 129D99E4
64CAB820 847EDAD9 DF0B4E4C 73A05DD2
BD62A8A9 FA603DD2 E2A8A6F8 98F76E28
D58AD221 B583D7A4 71020301 0001
quit
exit
  addressed-key 10.1.1.2 encryption
  key-string
00302017 4A7D385B 1234EF29 335FC973
2DD50A37 C4F4B0FD 9DADE748 429618D5
18242BA3 2EDFBDD3 4296142A DDF7D3D8
08407685 2F2190A0 0B43F1BD 9A8A26DB
07953829 791FCDE9 A98420F0 6A82045B
90288A26 DBC64468 7789F76E EE21
quit
  exit
  addressed-key 10.1.1.2 signature
  key-string
0738BC7A 2BC3E9F0 679B00FE 098533AB
01030201 42DD06AF E228D24C 458AD228
58BB5DDD F4836401 2A2D7163 219F882E
64CE69D4 B583748A 241BED0F 6E7F2F16
0DE0986E DF02031F 4B0B0912 F68200C4
C625C389 0BFF3321 A2598935 C1B1
quit
  exit
  exit

```

Related Commands

Command	Description
address	Specifies the IP address of the remote RSA public key of the remote peer you will manually configure.
addressed-key	Specifies the RSA public key of the peer you will manually configure.
crypto key pubkey-chain rsa	Enters public key configuration mode (to allow you to manually specify the RSA public keys of other devices).
key-string (IKE)	Specifies the RSA public key of a remote peer.
show crypto key pubkey-chain rsa	Displays peer RSA public keys stored on your router.

no crypto xauth

To ignore extended authentication (Xauth) during an Internet Key Exchange (IKE) Phase 1 negotiation, use the **no crypto xauth** command in global configuration mode. To consider Xauth proposals, use the **crypto xauth** command.

no crypto xauth *interface*

crypto xauth *interface*

Syntax Description

<i>interface</i>	Interface whose IP address is the local endpoint to which the remote peer will send IKE requests.
------------------	---

Defaults

No default behaviors or values

Command Modes

Global configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.

Usage Guidelines

The **no** version of this command was introduced to support Unity clients that do not require Xauth when using Internet Security Association and Key Management Protocol (ISAKMP) profiles.

Examples

The following example shows that Xauth proposals on Ethernet 1/1 are to be ignored:

```
no crypto xauth Ethernet1/1
```

no ip inspect

To turn off Context-based Access Control (CBAC) completely at a firewall, use the **no ip inspect** command in global configuration mode.

no ip inspect

Syntax Description This command has no arguments or keywords.

Defaults No default behavior or values.

Command Modes Global configuration

Command History	Release	Modification
	11.2 P	This command was introduced.

Usage Guidelines Turn off CBAC with the **no ip inspect** global configuration command.



Note

The **no ip inspect** command removes all CBAC configuration entries and resets all CBAC global timeouts and thresholds to the defaults. All existing sessions are deleted and their associated access lists are removed.

Examples The following example turns off CBAC at a firewall:

```
no ip inspect
```

password (ca-trustpoint)

To specify the revocation password for the certificate, use the **password** command in ca-trustpoint configuration mode. To erase any stored passwords, use the **no** form of this command.

password *string*

no password

Syntax Description	<i>string</i>	Name of the password.
---------------------------	---------------	-----------------------

Defaults	You are prompted for the password during certificate enrollment.
-----------------	--

Command Modes	Ca-trustpoint configuration
----------------------	-----------------------------

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

Usage Guidelines	<p>Before you can issue the password command, you must enable the crypto ca trustpoint command, which declares the certification authority (CA) that your router should use and enters ca-trustpoint configuration mode.</p> <p>This command allows you to specify the revocation password for the certificate before actual certificate enrollment begins. The specified password is encrypted when the updated configuration is written to NVRAM by the router.</p> <p>If this command is enabled, you will not be prompted for a password during certificate enrollment.</p>
-------------------------	---

Examples	The following example shows how to specify the password “revokme” for the certificate request:
-----------------	--

```
crypto ca trustpoint frog
enrollment url http://frog.phoobin.com/
subject-name OU=Spiral Dept., O=tiedye.com
ip-address ethernet-0
auto-enroll regenerate
password revokme
```

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

password (line configuration)

To specify a password on a line, use the **password** command in line configuration mode. To remove the password, use the **no** form of this command.

password *password*

no password

Syntax Description	<i>password</i>	Character string that specifies the line password. The first character cannot be a number. The string can contain any alphanumeric characters, including spaces, up to 80 characters. You cannot specify the password in the format number-space-anything. The space after the number causes problems. For example, hello 21 is a legal password, but 21 hello is not. The password checking is case sensitive. For example, the password Secret is different than the password secret.
Defaults	No password is specified.	
Command Modes	Line configuration	
Command History	Release	Modification
	10.0	This command was introduced.
Usage Guidelines	When an EXEC process is started on a line with password protection, the EXEC prompts for the password. If the user enters the correct password, the EXEC prints its normal privileged prompt. The user can try three times to enter a password before the EXEC exits and returns the terminal to the idle state.	
Examples	The following example removes the password from virtual terminal lines 1 to 4: <pre>line vty 1 4 no password</pre>	
Related Commands	Command	Description
	enable password	Sets a local password to control access to various privilege levels.

permit (reflexive)

To create a reflexive access list and to enable its temporary entries to be automatically generated, use the **permit** command in access-list configuration mode. To delete the reflexive access list (if only one protocol was defined) or to delete protocol entries from the reflexive access list (if multiple protocols are defined), use the **no** form of this command.

permit *protocol source source-wildcard destination destination-wildcard reflect name [timeout seconds]*

no permit *protocol source-wildcard destination destination-wildcard reflect name*

Syntax Description	
<i>protocol</i>	Name or number of an IP protocol. It can be one of the keywords gre , icmp , ip , ipinip , nos , 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, Transmission Control Protocol, and User Datagram Protocol), use the keyword ip .
<i>source</i>	Number of the network or host from which the packet is being sent. There are three other ways to specify the source: <ul style="list-style-type: none"> • Use a 32-bit quantity in four-part, dotted-decimal format. • Use the keyword any as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. This keyword is normally not recommended (see the section “Usage Guidelines”). • Use host source as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of source 0.0.0.0.
<i>source-wildcard</i>	Wildcard bits (mask) to be applied to source. There are three other ways to specify the source wildcard: <ul style="list-style-type: none"> • Use a 32-bit quantity in four-part, dotted-decimal format. Place ones in the bit positions you want to ignore. • Use the keyword any as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. This keyword is normally not recommended (see the section “Usage Guidelines”). • Use host source as an abbreviation for a <i>source</i> and <i>source-wildcard</i> of source 0.0.0.0.
<i>destination</i>	Number of the network or host to which the packet is being sent. There are three other ways to specify the destination: <ul style="list-style-type: none"> • Use a 32-bit quantity in four-part, dotted-decimal format. • Use the keyword any as an abbreviation for the <i>destination</i> and <i>destination-wildcard</i> of 0.0.0.0 255.255.255.255. This keyword is normally not recommended (see the section “Usage Guidelines”). • Use host destination as an abbreviation for a <i>destination</i> and <i>destination-wildcard</i> of destination 0.0.0.0.

<i>destination-wildcard</i>	Wildcard bits to be applied to the destination. There are three other ways to specify the destination wildcard: <ul style="list-style-type: none"> • Use a 32-bit quantity in four-part, dotted-decimal format. Place ones in the bit positions you want to ignore. • Use the keyword any as an abbreviation for a <i>destination</i> and <i>destination-wildcard</i> of 0.0.0.0 255.255.255.255. This keyword is normally <i>not</i> recommended (see the section “Usage Guidelines”). • Use host destination as an abbreviation for a <i>destination</i> and <i>destination-wildcard</i> of <i>destination</i> 0.0.0.0.
reflect	Identifies this access list as a reflexive access list.
<i>name</i>	Specifies the name of the reflexive access list. Names cannot contain a space or quotation mark, and must begin with an alphabetic character to prevent ambiguity with numbered access lists. The name can be up to 64 characters long.
timeout seconds	(Optional) Specifies the number of seconds to wait (when no session traffic is being detected) before entries expire in this reflexive access list. Use a positive integer from 0 to $2^{32}-1$. If not specified, the number of seconds defaults to the global timeout value.

Defaults

If this command is not configured, no reflexive access lists will exist, and no session filtering will occur. If this command is configured without specifying a **timeout** value, entries in this reflexive access list will expire after the global timeout period.

Command Modes

Access-list configuration

Command History

Release	Modification
11.3	This command was introduced.

Usage Guidelines

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

For this command to work, you must also nest the reflexive access list using the **evaluate** command.

This command creates a reflexive access list and triggers the creation of entries in the same reflexive access list. This command must be an entry (condition statement) in an extended named IP access list.

If you are configuring reflexive access lists for an external interface, the extended named IP access list should be one which is applied to outbound traffic.

If you are configuring reflexive access lists for an internal interface, the extended named IP access list should be one which is applied to inbound traffic.

IP sessions that originate from within your network are initiated with a packet exiting your network. When such a packet is evaluated against the statements in the extended named IP access list, the packet is also evaluated against this reflexive **permit** entry.

As with all access list entries, the order of entries is important, because they are evaluated in sequential order. When an IP packet reaches the interface, it will be evaluated sequentially by each entry in the access list until a match occurs.

If the packet matches an entry prior to the reflexive **permit** entry, the packet will not be evaluated by the reflexive **permit** entry, and no temporary entry will be created for the reflexive access list (session filtering will not be triggered).

The packet will be evaluated by the reflexive **permit** entry if no other match occurs first. Then, if the packet matches the protocol specified in the reflexive **permit** entry, the packet is forwarded and a corresponding temporary entry is created in the reflexive access list (unless the corresponding entry already exists, indicating the packet belongs to a session in progress). The temporary entry specifies criteria that permits traffic into your network only for the same session.

Characteristics of Reflexive Access List Entries

This command enables the creation of temporary entries in the same reflexive access list that was defined by this command. The temporary entries are created when a packet exiting your network matches the protocol specified in this command. (The packet “triggers” the creation of a temporary entry.) These entries have the following characteristics:

- The entry is a **permit** entry.
- The entry specifies the same IP upper-layer protocol as the original triggering packet.
- The entry specifies the same source and destination addresses as the original triggering packet, except the addresses are swapped.
- If the original triggering packet is TCP or UDP, the entry specifies the same source and destination port numbers as the original packet, except the port numbers are swapped.

If the original triggering packet is a protocol other than TCP or UDP, port numbers do not apply, and other criteria are specified. For example, for ICMP, type numbers are used: the temporary entry specifies the same type number as the original packet (with only one exception: if the original ICMP packet is type 8, the returning ICMP packet must be type 0 to be matched).

- The entry inherits all the values of the original triggering packet, with exceptions only as noted in the previous four bullets.
- IP traffic entering your internal network will be evaluated against the entry, until the entry expires. If an IP packet matches the entry, the packet will be forwarded into your network.
- The entry will expire (be removed) after the last packet of the session is matched.
- If no packets belonging to the session are detected for a configurable length of time (the timeout period), the entry will expire.

Examples

The following example defines a reflexive access list *tcptraffic*, in an outbound access list that permits all Border Gateway Protocol and Enhanced Interior Gateway Routing Protocol traffic and denies all ICMP traffic. This example is for an external interface (an interface connecting to an external network).

First, the interface is defined and the access list is applied to the interface for outbound traffic.

```
interface Serial 1
  description Access to the Internet via this interface
  ip access-group outboundfilters out
```

Next, the outbound access list is defined and the reflexive access list *tcptraffic* is created with a reflexive **permit** entry.

```
ip access-list extended outboundfilters
  permit tcp any any reflect tcptraffic
```

Related Commands

Command	Description
evaluate	Nests a reflexive access list within an access list.
ip access-list	Defines an IP access list by name.
ip reflexive-list timeout	Specifies the length of time that reflexive access list entries will continue to exist when no packets in the session are detected.

pool (isakmp-group)

To define a local pool address, use the **pool** command in Internet Security Association Key Management Protocol (ISAKMP) group configuration mode. To remove a local pool from your configuration, use the **no** form of this command.

pool *name*

no pool *name*

Syntax Description

<i>name</i>	Name of the local pool address.
-------------	---------------------------------

Defaults

No default behavior or values.

Command Modes

ISAKMP group configuration

Command History

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

Usage Guidelines

Use the **pool** command to refer to an IP local pool address, which defines a range of addresses that will be used to allocate an internal IP address to a client. Although a user must define at least one pool name, a separate pool may be defined for each group policy.



Note

This command must be defined and refer to a valid IP local pool address, or the client connection will fail.

Examples

The following example shows how to refer to the local pool address “dog”:

```
crypto isakmp client configuration group cisco
  key cisco
  dns 2.2.2.2 2.3.2.3
  pool dog
  acl 199
!
ip local pool dog 10.1.1.1 10.1.1.254
```

Related Commands

Command	Description
crypto isakmp client configuration group	Specifies which group’s policy profile will be defined.
ip local pool	Configures a local pool of IP addresses to be used when a remote peer connects to a point-to-point interface.