CHAPTER 23

# queue-limit through rtp-conformance Commands

# queue-limit (priority-queue)

To specify the depth of the priority queues, use the **queue-limit** command in priority-queue mode. To remove this specification, use the **no** form of this command.

**queue-limit** *number-of-packets*

**no queue-limit** *number-of-packets*

## Syntax Description

| | |
|---|---|
| *number-of-packets* | Specifies the maximum number of low-latency or normal priority packets that can be queued (that is, buffered) before the interface begins dropping packets. See the Usage Notes section for the range of possible values. |

## Defaults

The default queue limit is 1024 packets.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Priority-queue | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Usage Guidelines

The security appliance allows two classes of traffic: low-latency queuing (LLQ) for higher priority, latency sensitive traffic (such as voice and video) and best-effort, the default, for all other traffic. The security appliance recognizes priority traffic and enforces appropriate Quality of Service (QoS) policies. You can configure the size and depth of the priority queue to fine-tune the traffic flow.

You must use the **priority-queue** command to create the priority queue for an interface before priority queuing takes effect. You can apply one **priority-queue** command to any interface that can be defined by the **nameif** command.

The **priority-queue** command enters priority-queue mode, as shown by the prompt. In priority-queue mode, you can configure the maximum number of packets allowed in the transmit queue at any given time (**tx-ring-limit** command) and the number of packets of either type (priority or best -effort) allowed to be buffered before dropping packets (**queue-limit** command).

**Note** You *must* configure the **priority-queue** command in order to enable priority queueing for the interface.

The tx-ring-limit and the queue-limit that you specify affect both the higher priority low-latency queue and the best-effort queue. The tx-ring-limit is the number of either type of packets allowed into the driver before the driver pushes back to the queues sitting in front of the interface to let them buffer packets until the congestion clears. In general, you can adjust these two parameters to optimize the flow of low-latency traffic.

Because queues are not of infinite size, they can fill and overflow. When a queue is full, any additional packets cannot get into the queue and are dropped. This is *tail drop*. To avoid having the queue fill up, you can use the **queue-limit** command to increase the queue buffer size.

**Note** The upper limit of the range of values for the **queue-limit** and **tx-ring-limit** commands is determined dynamically at run time. To view this limit, enter **help** or **?** on the command line. The key determinant is the memory needed to support the queues and the memory available on the device. The queues must not exceed the available memory. The theoretical maximum number of packets is 2147483647.

On ASA Model 5505 (only), configuring priority-queue on one interface overwrites the same configuration on all other interfaces. That is, only the last applied configuration is present on all interfaces. Further, if the priority-queue configuration is removed from one interface, it is removed from all interfaces.

To work around this issue, configure the **priority-queue** command on only one interface. If different interfaces need different settings for the **queue-limit** and/or **tx-ring-limit** commands, use the largest of all queue-limits and smallest of all tx-ring-limits on any one interface (CSCsi13132).

## Examples

The following example configures a priority queue for the interface named test, specifying a queue limit of 30,000 packets and a transmit queue limit of 256 packets.

```
hostname(config)# priority-queue test
hostname(priority-queue)# queue-limit 30000
hostname(priority-queue)# tx-ring-limit 256
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure priority-queue** | Removes the current priority queue configuration on the named interface. |
| **priority-queue** | Configures priority queuing on an interface. |
| **show priority-queue statistics** | Shows the priority-queue statistics for the named interface. |
| **show running-config [all] priority-queue** | Shows the current priority queue configuration. If you specify the **all** keyword, this command displays all the current priority queue, queue-limit, and tx-ring-limit configuration values. |
| **tx-ring-limit** | Sets the maximum number of packets that can be queued at any given time in the Ethernet transmit driver. |

# queue-limit (tcp-map)

To configure the maximum number of out-of-order packets that can be buffered and put in order for a TCP connection, use the **queue-limit** command in tcp-map configuration mode. To set the value back to the default, use the **no** form of this command. This command is part of the TCP normalization policy enabled using the **set connection advanced-options** command.

**queue-limit** *pkt_num* [**timeout** *seconds*]

**no queue-limit**

## Syntax Description

| | |
|---|---|
| *pkt_num* | Specifies the maximum number of out-of-order packets that can be buffered and put in order for a TCP connection, between 1 and 250. The default is 0, which means this setting is disabled and the default system queue limit is used depending on the type of traffic. See the "Usage Guidelines" section for more information. |
| **timeout** *seconds* | (Optional) Sets the maximum amount of time that out-of-order packets can remain in the buffer, between 1 and 20 seconds. The default is 4 seconds. If packets are not put in order and passed on within the timeout period, then they are dropped. You cannot change the timeout for any traffic if the *pkt_num* argument is set to 0; you need to set the limit to be 1 or above for the **timeout** keyword to take effect. |

## Defaults

The default setting is 0, which means this command is disabled.

The default timeout is 4 seconds.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Tcp-map configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |
| 7.2(4)/8.0(4) | The **timeout** keyword was added. |

## Usage Guidelines

To enable TCP normalization, use the Modular Policy Framework:

1. **tcp-map**—Identifies the TCP normalization actions.
   a. **queue-limit**—In tcp-map configuration mode, you can enter the **queue-limit** command and many others.

2. **class-map**—Identify the traffic on which you want to perform TCP normalization.
3. **policy-map**—Identify the actions associated with each class map.
   a. **class**—Identify the class map on which you want to perform actions.
   b. **set connection advanced-options**—Identify the tcp-map you created.
4. **service-policy**—Assigns the policy map to an interface or globally.

If you do not enable TCP normalization, or if the **queue-limit** command is set to the default of 0, which means it is disabled, then the default system queue limit is used depending on the type of traffic:

- Connections for application inspection (the **inspect** command), IPS (the **ips** command), and TCP check-retransmission (the TCP map **check-retransmission** command) have a queue limit of 3 packets. If the security appliance receives a TCP packet with a different window size, then the queue limit is dynamically changed to match the advertized setting.
- For other TCP connections, out-of-order packets are passed through untouched.

If you set the **queue-limit** command to be 1 or above, then the number of out-of-order packets allowed for all TCP traffic matches this setting. For example, for application inspection, IPS, and TCP check-retransmission traffic, any advertised settings from TCP packets are ignored in favor of the **queue-limit** setting. For other TCP traffic, out-of-order packets are now buffered and put in order instead of passed through untouched.

## Examples

The following example sets the queue limit to 8 packets and the buffer timeout to 6 seconds for all Telnet connections:

```
hostname(config)# tcp-map tmap
hostname(config-tcp-map)# queue-limit 8 timeout 6
hostname(config)# class-map cmap
hostname(config-cmap)# match port tcp eq telnet
hostname(config)# policy-map pmap
hostname(config-pmap)# class cmap
hostname(config-pmap)# set connection advanced-options tmap
hostname(config)# service-policy pmap global
hostname(config)#
```

## Related Commands

| Command | Description |
|---|---|
| **class-map** | Identifies traffic for a service policy. |
| **policy-map** | dentifies actions to apply to traffic in a service policy. |
| **set connection advanced-options** | Enables TCP normalization. |
| **service-policy** | Applies a service policy to interface(s). |
| **show running-config tcp-map** | Shows the TCP map configuration. |
| **tcp-map** | Creates a TCP map and allows access to tcp-map configuration mode. |

# quit

To exit the current configuration mode, or to logout from privileged or user EXEC modes, use the **quit** command.

**quit**

## Syntax Description

This command has no arguments or keywords.

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| User EXEC | • | • | • | • | • |

## Command History

| Release | Modification |
|---|---|
| Preexisting | This command was preexisting. |

## Usage Guidelines

You can also use the key sequence **Ctrl Z** to exit global configuration (and higher) modes. This key sequence does not work with privileged or user EXEC modes.

When you enter the **quit** command in privileged or user EXEC modes, you log out from the security appliance. Use the **disable** command to return to user EXEC mode from privileged EXEC mode.

## Examples

The following example shows how to use the **quit** command to exit global configuration mode, and then logout from the session:

```
hostname(config)# quit
hostname# quit

Logoff
```

The following example shows how to use the **quit** command to exit global configuration mode, and then use the **disable** command to exit privileged EXEC mode:

```
hostname(config)# quit
hostname# disable
hostname>
```

**Related Commands**

| Command | Description |
|---|---|
| **exit** | Exits a configuration mode or logs out from privileged or user EXEC modes. |

# radius-common-pw

To specify a common password to be used for all users who are accessing this RADIUS authorization server through this security appliance, use the **radius-common-pw** command in AAA-server host mode. To remove this specification, use the **no** form of this command:

**radius-common-pw** *string*

**no radius-common-pw**

## Syntax Description

| | |
|---|---|
| *string* | A case-sensitive, alphanumeric keyword of up to 127 characters to be used as a common password for all authorization transactions with this RADIUS server. |

## Defaults

No default behaviors or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| AAA-server host | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | Introduced in this release. |

## Usage Guidelines

This command is valid only for RADIUS authorization servers.

The RADIUS authorization server requires a password and username for each connecting user. The security appliance provides the username automatically. You enter the password here. The RADIUS server administrator must configure the RADIUS server to associate this password with each user authorizing to the server via this security appliance. Be sure to provide this information to your RADIUS server administrator.

If you do not specify a common user password, each user's password is his or her own username. For example, a user with the username "jsmith" would enter "jsmith". If you are using usernames for the common user passwords, as a security precaution do not use this RADIUS server for authorization anywhere else on your network.

**Note** This field is essentially a space-filler. The RADIUS server expects and requires it, but does not use it. Users do not need to know it.

## Examples

The following example configures a RADIUS AAA server group named “svrgrp1” on host “1.2.3.4”, sets the timeout interval to 9 seconds, sets the retry interval to 7 seconds, and configures the RADIUS commnon password as “allauthpw”.

```
hostname(config)# aaa-server svrgrp1 protocol radius
hostname(config-aaa-server-group)# aaa-server svrgrp1 host 1.2.3.4
hostname(config-aaa-server-host)# timeout 9
hostname(config-aaa-server-host)# retry 7
hostname(config-aaa-server-host)# radius-common-pw allauthpw
hostname(config-aaa-server-host)# exit
hostname(config)#
```

## Related Commands

| Command | Description |
|---|---|
| **aaa-server host** | Enter AAA server host configuration mode so you can configure AAA server parameters that are host-specific. |
| **clear configure aaa-server** | Remove all AAA command statements from the configuration. |
| **show running-config aaa-server** | Displays AAA server statistics for all AAA servers, for a particular server group, for a particular server within a particular group, or for a particular protocol |

# radius-reject-message

To enable the display of a RADIUS reject message on the login screen when authentication is rejected, use the **radius-eject-message** command from tunnel-group webvpn attributes configuration mode. To remove the command from the configuration, use the **no** form of the command:

**radius-reject-message**

**no radius-reject-message**

## Defaults

The default is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| Command Mode | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| | Routed | Transparent | Single | Context | System |
| Tunnel-group webvpn configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Enable this command if you want to display to remote users a RADIUS message about an authentication failure.

## Examples

The following example enables the display of a RADIUS rejection message for the connection profile named engineering:

```
hostname(config)# tunnel-group engineering webvpn-attributes
hostname(config-tunnel-webvpn)# radius-reject-message
```

# radius-with-expiry (removed)

To have the security appliance use MS-CHAPv2 to negotiate a password update with the user during authentication, use the **radius-with-expiry** command in tunnel-group ipsec-attributes configuration mode. The security appliance ignores this command if RADIUS authentication has not been configured. To return to the default value, use the **no** form of this command.

**radius-with-expiry**

**no radius-with-expiry**

## Syntax Description

This command has no arguments or keywords.

## Defaults

The default setting for this command is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Tunnel-group ipsec-attributes configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |
| 7.1(1) | This command was deprecated. The **password-management** command replaces it. The **no** form of the **radius-with-expiry** command is no longer supported. |
| 8.0(2) | This command was deprecated. |

## Usage Guidelines

You can apply this attribute only to IPSec remote-access tunnel-group type.

## Examples

The following example entered in config-ipsec configuration mode, configures Radius with Expiry for the remote-access tunnel group named remotegrp:

```
hostname(config)# tunnel-group remotegrp type ipsec_ra
hostname(config)# tunnel-group remotegrp ipsec-attributes
hostname(config-tunnel-ipsec)# radius-with-expiry
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure tunnel-group** | Clears all configured tunnel groups. |
| **password-management** | Enables password management. This command, in the tunnel-group general-attributes configuration mode, replaces the **radius-with-expiry** command. |
| **show running-config tunnel-group** | Shows the indicated certificate map entry. |
| **tunnel-group ipsec-attributes** | Configures the tunnel-group ipsec-attributes for this group. |

# ras-rcf-pinholes

To enable call setup between H.323 endpoints when the Gatekeeper is inside the network, use the **ras-rcf-pinholes** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

**ras-rcf-pinholes enable**

**no ras-rcf-pinholes enable**

## Syntax Description

| | |
|---|---|
| **enable** | Enables call setup between H.323 endpoints. |

## Defaults

By default, this option is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Parameters configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(5) | This command was introduced. |

## Usage Guidelines

The security appliance includes options to open pinholes for calls based on the RegistrationRequest/RegistrationConfirm (RRQ/RCF) messages. Because these RRQ/RCF messages are sent to and from the Gatekeeper, the calling endpoint's IP address is unknown and the security appliance opens a pinhole through source IP address/port 0/0.

## Examples

The following example shows how to set up an action for protocol violation in a policy map:

```
hostname(config)# policy-map type inspect h323 h323_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# ras-rcf-pinholes enable
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Identifies a class map name in the policy map. |
| **class-map type inspect** | Creates an inspection class map to match traffic specific to an application. |

| Command | Description |
|---|---|
| **policy-map** | Creates a Layer 3/4 policy map. |
| **show running-config policy-map** | Display all current policy map configurations. |

# rate-limit

When using the Modular Policy Framework, limit the rate of messages for packets that match a **match** command or class map by using the **rate-limit** command in match or class configuration mode. This rate limit action is available in an inspection policy map (the **policy-map type inspect** command) for application traffic; however, not all applications allow this action. To disable this action, use the **no** form of this command.

**rate-limit** *messages_per_second*

**no rate-limit** *messages_per_second*

## Syntax Description

| | |
|---|---|
| *messages_per_second* | Limits the messages per second. |

## Defaults

No default behaviors or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Match and class configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

An inspection policy map consists of one or more **match** and **class** commands. The exact commands available for an inspection policy map depends on the application. After you enter the **match** or **class** command to identify application traffic (the **class** command refers to an existing **class-map type inspect** command that in turn includes **match** commands), you can enter the **rate-limit** command to limit the rate of messages.

When you enable application inspection using the **inspect** command in a Layer 3/4 policy map (the **policy-map** command), you can enable the inspection policy map that contains this action, for example, enter the **inspect dns dns_policy_map** command where dns_policy_map is the name of the inspection policy map.

## Examples

The following example limits the invite requests to 100 messages per second:

```
hostname(config-cmap)# policy-map type inspect sip sip-map1
hostname(config-pmap-c)# match request-method invite
hostname(config-pmap-c)# rate-limit 100
```

## Related Commands

| Commands | Description |
|---|---|
| **class** | Identifies a class map name in the policy map. |
| **class-map type inspect** | Creates an inspection class map to match traffic specific to an application. |
| **policy-map** | Creates a Layer 3/4 policy map. |
| **policy-map type inspect** | Defines special actions for application inspection. |
| **show running-config policy-map** | Display all current policy map configurations. |

# reactivation-mode

To specify the method by which failed servers in a group are reactivated, use the **reactivation-mode** command in aaa-server protocol mode. To remove this specification, use the **no** form of this command:

**reactivation-mode** {**depletion** [**deadtime** *minutes*] | **timed**}

**no reactivation-mode** [**depletion** [**deadtime** *minutes*] | **timed**]

## Syntax Description

| | |
|---|---|
| **deadtime** *minutes* | (Optional) Specifies the amount of time in minutes, between 0 and 1440, that elapses between the disabling of the last server in the group and the subsequent re-enabling of all servers. The default is 10 minutes. |
| **depletion** | Reactivates failed servers only after all of the servers in the group are inactive. |
| **timed** | Reactivates failed servers after 30 seconds of down time. |

## Defaults

The default reactivation mode is depletion, and the default deadtime value is 10.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Aaa-server protcocol configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Usage Guidelines

Each server group has an attribute that specifies the reactivation policy for its servers.

In **depletion** mode, when a server is deactivated, it remains inactive until all other servers in the group are inactive. When and if this occurs, all servers in the group are reactivated. This approach minimizes the occurrence of connection delays due to failed servers. When **depletion** mode is in use, you can also specify the **deadtime** parameter. The **deadtime** parameter specifies the amount of time (in minutes) that will elapse between the disabling of the last server in the group and the subsequent re-enabling of all servers. This parameter is meaningful only when the server group is being used in conjunction with the local fallback feature.

In **timed** mode, failed servers are reactivated after 30 seconds of down time. This is useful when customers use the first server in a server list as the primary server and prefer that it is online whenever possible. This policy breaks down in the case of UDP servers. Since a connection to a UDP server will

not fail, even if the server is not present, UDP servers are put back on line blindly. This could lead to slowed connection times or connection failures if a server list contains multiple servers that are not reachable.

Accounting server groups that have simultaneous accounting enabled are forced to use the **timed** mode. This implies that all servers in a given list are equivalent.

## Examples

The following example configures aTACACS+ AAA server named "srvgrp1" to use the depletion reactivation mode, with a deadtime of 15 minutes:

```
hostname(config)# aaa-server svrgrp1 protocol tacacs+
hostname(config-aaa-sersver-group)# reactivation-mode depletion deadtime 15
hostname(config-aaa-server)# exit
hostname(config)#
```

The following example configures aTACACS+ AAA server named "srvgrp1" to use timed reactivation mode:

```
hostname(config)# aaa-server svrgrp2 protocol tacacs+
hostname(config-aaa-server)# reactivation-mode timed
hostname(config-aaa-server)#
```

## Related Commands

| | |
|---|---|
| **accounting-mode** | Indicates whether accounting messages are sent to a single server or sent to all servers in the group. |
| **aaa-server protocol** | Enters AAA server group configuration mode so you can configure AAA server parameters that are group-specific and common to all hosts in the group. |
| **max-failed-attempts** | Specifies the number of failures that will be tolerated for any given server in the server group before that server is deactivated. |
| **clear configure aaa-server** | Removes all AAA server configuration. |
| **show running-config aaa-server** | Displays AAA server statistics for all AAA servers, for a particular server group, for a particular server within a particular group, or for a particular protocol |

# record-entry

To specify the trustpoints to be used for the creation of the CTL file, use the record-entry command in ctl-file configuration mode. To remove a record entry from a CTL, use the **no** form of this command.

**record-entry** [ **capf** | **cucm** | **cucm-tftp** | **tftp** ] **trustpoint** *trustpoint* **address** *ip_address* [**domain-name** *domain_name*]

**no record-entry** [ **capf** | **cucm** | **cucm-tftp** | **tftp** ] **trustpoint** *trust_point* **address** *ip_address* [**domain-name** *domain_name*]

## Syntax Description

| | |
|---|---|
| **capf** | Specifies the role of this trustpoint to be CAPF. Only one CAPF trustpoint can be configured. |
| **cucm** | Specifies the role of this trustpoint to be CCM. Multiple CCM trustpoints can be configured. |
| **cucm-tftp** | Specifies the role of this trustpoint to be CCM+TFTP. Multiple CCM+TFTP trustpoints can be configured. |
| **domain-name** *domain_name* | (Optional) Specifies the domain name of the trustpoint used to create the DNS field for the trustpoint. This is appended to the Common Name field of the Subject DN to create the DNS Name. The domain name should be configured when the FQDN is not configured for the trustpoint. |
| **address** *ip_address* | Specifies the IP address of the trustpoint. |
| **tftp** | Specifies the role of this trustpoint to be TFTP. Multiple TFTP trustpoints can be configured. |
| **trustpoint** *trust_point* | Sets the name of the trustpoint installed. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| CTL-file configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(4) | The command was introduced. |

## Usage Guidelines

Only one domain-name can be specified. If the CTL file does not exist, manually export this certificate from CUCM to the security appliance.

Use this command only when you have not configured a CTL file for the Phone Proxy. Do not use this command when you have already configured a CTL file.

The IP address you specify in the *ip_address* argument must be the global address or address as seen by the IP phones because it will be the IP address used for the CTL record for the trustpoint.

Add additional record-entry configurations for each entity that is required in the CTL file.

## Examples

The following example shows the use of the **record-entry** command to specify the trustpoints to be used for the creation of the CTL file:

```
hostname(config-ctl-file)# record-entry cucm-tftp trustpoint cucm1 address 192.168.1.2
```

## Related Commands

| Command | Description |
|---|---|
| **ctl-file (global)** | Specifies the CTL file to create for Phone Proxy configuration or the CTL file to parse from Flash memory. |
| **ctl-file (phone-proxy)** | Specifies the CTL file to use for Phone Proxy configuration. |
| **phone-proxy** | Configures the Phone Proxy instance. |

# redirect-fqdn

To enable or disable redirection using a fully-qualified domain name in vpn load-balancing mode, use the **redirect-fqdn enable** command in global configuration mode.

**redirect-fqdn {enable | disable}**

**no redirect-fqdn {enable | disable}**

**Note** To use VPN load balancing, you must have an ASA Model 5510 with a Plus license or an ASA Model 5520 or higher. VPN load balancing also requires an active 3DES/AES license. The security appliance checks for the existence of this crypto license before enabling load balancing. If it does not detect an active 3DES or AES license, the security appliance prevents the enabling of load balancing and also prevents internal configuration of 3DES by the load balancing system unless the license permits this usage.

## Syntax Description

| | |
|---|---|
| **disable** | Disables redirection with fully-qualified domain names. |
| **enable** | Enables redirection with fully-qualified domain names. |

## Defaults

This behavior is disabled by default.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Vpn load-balancing mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

By default, the ASA sends only IP addresses in load-balancing redirection to a client. If certificates are in use that are based on DNS names, the certificates will be invalid when redirected to a secondary device.

As a VPN cluster master, this security appliance can send a fully qualified domain name (FQDN), using reverse DNS lookup, of a cluster device (another security appliance in the cluster), instead of its outside IP address, when redirecting VPN client connections to that cluster device.

All of the outside and inside network interfaces on the load-balancing devices in a cluster must be on the same IP network.

To do WebVPN load Balancing using FQDNs rather than IP addresses, you must do the following configuration steps:

**Step 1** Enable the use of FQDNs for Load Balancing with the **redirect-fqdn enable** command.

**Step 2** Add an entry for each of your ASA outside interfaces into your DNS server, if such entries are not already present. Each ASA outside IP address should have a DNS entry associated with it for lookups. These DNS entries must also be enabled for Reverse Lookup.

**Step 3** Enable DNS lookups on your ASA with the command - "dns domain-lookup inside" (or whichever interface has a route to your DNS server).

**Step 4** Define your DNS server IP address on the ASA; for example: `dns name-server 10.2.3.4` (IP address of your DNS server)

## Examples

The following is an example of the **redirect-fqdn** command that disables redirection:

```
hostname(config)# vpn load-balancing
hostname(config-load-balancing)# redirect-fqdn disable
hostname(config-load-balancing)#
```

The following is an example of a VPN load-balancing command sequence that includes an interface command that enables redirection for a fully-qualified domain name, specifies the public interface of the cluster as "test" and the private interface of the cluster as "foo":

```
hostname(config)# interface GigabitEthernet 0/1
hostname(config-if)# ip address 209.165.202.159 255.255.255.0
hostname(config)# nameif test
hostname(config)# interface GigabitEthernet 0/2
hostname(config-if)# ip address 209.165.201.30 255.255.255.0
hostname(config)# nameif foo
hostname(config)# vpn load-balancing
hostname(config-load-balancing)# nat 192.168.10.10
hostname(config-load-balancing)# priority 9
hostname(config-load-balancing)# interface lbpublic test
hostname(config-load-balancing)# interface lbprivate foo
hostname(config-load-balancing)# cluster ip address 209.165.202.224
hostname(config-load-balancing)# cluster key 123456789
hostname(config-load-balancing)# cluster encryption
hostname(config-load-balancing)# cluster port 9023
hostname(config-load-balancing)# redirect-fqdn enable
hostname(config-load-balancing)# participate
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure vpn load-balancing** | Removes the load-balancing runtime configuration and disables load balancing. |
| **show running-config vpn load-balancing** | Displays the the current VPN load-balancing virtual cluster configuration. |
| **show vpn load-balancing** | Displays VPN load-balancing runtime statistics. |
| **vpn load-balancing** | Enters vpn load-balancing mode. |

# redistribute (EIGRP)

To redistribute routes from one routing domain into the EIGRP routing process, use the **redistribute** command in router configuration mode. To remove the redistribution, use the **no** form of this command.

**redistribute** {{**ospf** *pid* [**match** {**internal** | **external** [**1** | **2**] | **nssa-external** [**1** | **2**]}]} | **rip** | **static** | **connected**} [**metric** *bandwidth delay reliability load mtu*] [**route-map** *map_name*]

**no redistribute** {{**ospf** *pid* [**match** {**internal** | **external** [**1** | **2**] | **nssa-external** [**1** | **2**]}]} | **rip** | **static** | **connected**} [**metric** *bandwidth delay reliability load mtu*] [**route-map** *map_name*]

## Syntax Description

| | |
|---|---|
| *bandwidth* | EIGRP bandwidth metric in Kilobits per second. Valid values are from 1 to 4294967295. |
| **connected** | Specifies redistributing a network connected to an interface into the EIGRP routing process. |
| *delay* | EIGRP delay metric, in 10 microsecond units. Valid values are from 0 to 4294967295. |
| **external** *type* | Specifies the OSPF metric routes that are external to a specified autonomous system; valid values are **1** or **2**. |
| **internal** *type* | Specifies OSPF metric routes that are internal to a specified autonomous system. |
| *load* | EIGRP effective bandwidth (loading) metric. Valid values are from 1 to 255, where 255 indicates 100% loaded. |
| **match** | (Optional) Specifies the conditions for redistributing routes from OSPF into EIGRP. |
| **metric** | (Optional) Specifies the values for the EIGRP metrics of routes redistributed into the EIGRP routing process. |
| *mtu* | The MTU of the path. Valid values are from 1 to 65535. |
| **nssa-external** *type* | Specifies the OSPF metric type for routes that are external to an NSSA; valid values are **1** or **2**. |
| **ospf** *pid* | Used to redistribute an OSPF routing process into the EIGRP routing process. The *pid* specifies the internally used identification parameter for an OSPF routing process; valid values are from 1 to 65535. |
| *reliability* | EIGRP reliability metric. Valid values are from 0 to 255, where 255 indicates 100% reliability. |
| **rip** | Specifies redistributing a network from the RIP routing process into the EIGRP routing process. |
| **route-map** *map_name* | (Optional) Name of the route map used to filter the imported routes from the source routing protocol to the EIGRP routing process. If not specified, all routes are redistributed. |
| **static** | Used to redistribute a static route into the EIGRP routing process. |

## Defaults

The following are the command defaults:

- **match**: **Internal**, **external 1**, **external 2**

## Command Modes

The following table shows the modes in which you can enter the command:

| Command Mode | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| | Routed | Transparent | Single | Context | System |
| Router configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

You must specify the **metric** with the redistribute command if you do not have a **default-metric** command in your EIGRP configuration.

## Examples

This example redistributes static and connected routes into the EIGRP routing process:

```
hostname(config)# router eigrp 100
hostname(config-router)# redistribute static
hostname(config-router)# redistribute connected
```

## Related Commands

| Command | Description |
|---|---|
| **router eigrp** | Creates an EIGRP routing process and enters configuration mode for that process. |
| **show running-config router** | Displays the commands in the global router configuration. |

# redistribute (OSPF)

To redistribute routes from one routing domain into an OSPF routing process, use the **redistribute** command in router configuration mode. To remove the redistribution, use the **no** form of this command.

**redistribute** {{**ospf** *pid* [**match** {**internal** | **external** [**1** | **2**] | **nssa-external** [**1** | **2**]}]} | **rip** | **static** | **connected** | **eigrp** *as-number*} [**metric** *metric_value*] [**metric-type** *metric_type*] [**route-map** *map_name*] [**tag** *tag_value*] [**subnets**]

**no redistribute** {{**ospf** *pid* [**match** {**internal** | **external** [**1** | **2**] | **nssa-external** [**1** | **2**]}]} | **rip** | **static** | **connected**} [**metric** *metric_value*] [**metric-type** *metric_type*] [**route-map** *map_name*] [**tag** *tag_value*] [**subnets**]

## Syntax Description

| | |
|---|---|
| **connected** | Specifies redistributing a network connected to an interface into an OSPF routing process. |
| **eigrp** *as-number* | Used to redistribute EIGRP routes into the OSPF routing process. The *as-number* specifies the autonomous system number of the EIGRP routing process. Valid values are from 1 to 65535. |
| **external** *type* | Specifies the OSPF metric routes that are external to a specified autonomous system; valid values are **1** or **2**. |
| **internal** *type* | Specifies OSPF metric routes that are internal to a specified autonomous system. |
| **match** | (Optional) Specifies the conditions for redistributing routes from one routing protocol into another. |
| **metric** *metric_value* | (Optional) Specifies the OSPF default metric value from 0 to 16777214. |
| **metric-type** *metric_type* | (Optional) The external link type associated with the default route advertised into the OSPF routing domain. It can be either of the following two values: **1** (Type 1 external route) or **2** (Type 2 external route). |
| **nssa-external** *type* | Specifies the OSPF metric type for routes that are external to an NSSA; valid values are **1** or **2**. |
| **ospf** *pid* | Used to redistribute an OSPF routing process into the current OSPF routing process. The *pid* specifies the internally used identification parameter for an OSPF routing process; valid values are from 1 to 65535. |
| **rip** | Specifies redistributing a network from the RIP routing process into the current OSPF routing process. |
| **route-map** *map_name* | (Optional) Name of the route map used to filter the imported routes from the source routing protocol to the current OSPF routing process. If not specified, all routes are redistributed. |
| **static** | Used to redistribute a static route into an OSPF process. |
| **subnets** | (Optional) For redistributing routes into OSPF, scopes the redistribution for the specified protocol. If not used, only classful routes are redistributed. |
| **tag** *tag_value* | (Optional) A 32-bit decimal value attached to each external route. This value is not used by OSPF itself. It may be used to communicate information between ASBRs. If none is specified, then the remote autonomous system number is used for routes from BGP and EGP; for other protocols, zero (0) is used. Valid values range from 0 to 4294967295. |

## Defaults

The following are the command defaults:

- **metric** *metric-value*: 0
- **metric-type** *type-value*: **2**
- **match**: **Internal**, **external 1**, **external 2**
- **tag** *tag-value*: 0

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Router configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| Preexisting | This command was preexisting. |
| 7.2(1) | This command was modified to include the **rip** keyword. |
| 8.0(2) | This command was modified to include the **eigrp** keyword. |

## Examples

This example shows how to redistribute static routes into the current OSPF process:

```
hostname(config)# router ospf 1
hostname(config-router)# redistribute static
```

## Related Commands

| Command | Description |
|---|---|
| redistribute (RIP) | Redistributes routes into the RIP routing process. |
| **router ospf** | Enters router configuration mode. |
| **show running-config router** | Displays the commands in the global router configuration. |

# redistribute (RIP)

To redistribute routes from another routing domain into the RIP routing process, use the **redistribute** command in router configuration mode. To remove the redistribution, use the **no** form of this command.

**redistribute** {{**ospf** *pid* [**match** {**internal** | **external** [**1** | **2**] | **nssa-external** [**1** | **2**]}]} | **static** | **connected** | **eigrp** *as-number*} [**metric** {*metric_value* | **transparent**}] [**route-map** *map_name*]

**no redistribute** {{**ospf** *pid* [**match** {**internal** | **external** [**1** | **2**] | **nssa-external** [**1** | **2**]}]} | **static** | **connected** | **eigrp** *as-number*} [**metric** {*metric_value* | **transparent**}] [**route-map** *map_name*]

## Syntax Description

| | |
|---|---|
| **connected** | Specifies redistributing a network connected to an interface into the RIP routing process. |
| **eigrp** *as-number* | Used to redistribute EIGRP routes into the RIP routing process. The *as-number* specifies the autonomous system number of the EIGRP routing process. Valid values are from 1 to 65535. |
| **external** *type* | Specifies the OSPF metric routes that are external to a specified autonomous system; valid values are **1** or **2**. |
| **internal** *type* | Specifies OSPF metric routes that are internal to a specified autonomous system. |
| **match** | (Optional) Specifies the conditions for redistributing routes from OSPF to RIP. |
| **metric** {*metric_value* \| **transparent**} | (Optional) Specifies the RIP metric value for the route being redistributed. Valid values for *metric_value* are from 0 to 16. Setting the metric to **transparent** causes the current route metric to be used. |
| **nssa-external** *type* | Specifies the OSPF metric type for routes that are external to a not-so-stubby area (NSSA); valid values are **1** or **2**. |
| **ospf** *pid* | Used to redistribute an OSPF routing process into the RIP routing process. The *pid* specifies the internally used identification parameter for an OSPF routing process; valid values are from 1 to 65535. |
| **route-map** *map_name* | (Optional) Name of the route map used to filter the imported routes from the source routing protocol to the RIP routing process. If not specified, all routes are redistributed. |
| **static** | Used to redistribute a static route into a RIP process. |

## Defaults

The following are the command defaults:

- **metric** *metric-value*: 0
- **match**: **Internal**, **external 1**, **external 2**

## Command Modes

The following table shows the modes in which you can enter the command:

| Command Mode | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| | Routed | Transparent | Single | Context | System |
| Router configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |
| 8.0(2) | This command was modified to include the **eigrp** keyword. |

## Examples

This example shows how to redistribute static routes into the current RIP process:

```
hostname(config)# router rip
hostname(config-router)# network 10.0.0.0
hostname(config-router)# redistribute static metric 2
```

## Related Commands

| Command | Description |
|---|---|
| **redistribute (EIGRP)** | Redistributes routes from other routing domains into EIGRP. |
| **redistribute (OSPF)** | Redistributes routes from other routing domains into OSPF. |
| **router rip** | Enables the RIP routing process and enters router configuration mode for that process. |
| **show running-config router** | Displays the commands in the global router configuration. |

# redundant-interface

To set which member interface of a redundant interface is active, use the **redundant-interface** command in privileged EXEC mode.

**redundant-interface redundant***number* **active-member** *physical_interface*

## Syntax Description

| | |
|---|---|
| **active-member** *physical_interface* | Sets the active member. See the **interface** command for accepted values. Both member interfaces must be the same physical type. |
| **redundant***number* | Specifies the redundant interface ID, such as **redundant1**. |

## Defaults

By default, the active interface is the first member interface listed in the configuration, if it is available.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC | • | • | • | — | • |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

To view which interface is active, enter the following command:

```
hostname# show interface redundantnumber detail | grep Member
```

For example:

```
hostname# show interface redundant1 detail | grep Member
      Members GigabitEthernet0/3(Active), GigabitEthernet0/2
```

## Examples

The following example creates a redundant interface. By default, gigabitethernet 0/0 is active because it is first in the configuration. The redundant-interface command sets gigabitethernet 0/1 as the active interface.

```
hostname(config-if)# interface redundant 1
hostname(config-if)# member-interface gigabitethernet 0/0
hostname(config-if)# member-interface gigabitethernet 0/1

hostname(config-if)# redundant-interface redundant1 active-member gigabitethernet0/1
```

## Related Commands

| Command | Description |
|---|---|
| **clear interface** | Clears counters for the **show interface** command. |
| **debug redundant-interface** | Displays debug messages related to redundant interface events or errors. |
| **interface redundant** | Creates a redundant interface. |
| **member-interface** | Assigns a member interface to a redundant interface pair. |
| **show interface** | Displays the runtime status and statistics of interfaces. |

# regex

To create a regular expression to match text, use the **regex** command in global configuration mode. To delete a regular expression, use the **no** form of this command.

**regex** *name regular_expression*

**no regex** *name* [*regular_expression*]

## Syntax Description

| | |
|---|---|
| *name* | Specifies the regular expression name, up to 40 characters in length. |
| *regular_expression* | Specifies the regular expression up to 100 characters in length. See "Usage Guidelines" for a list of metacharacters you can use in the regular expression. |

## Defaults

No default behaviors or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

The **regex** command can be used for various features that require text matching. For example, you can configure special actions for application inspection using Modular Policy Framework using an *inspection policy map* (see the **policy map type inspect** command). In the inspection policy map, you can identify the traffic you want to act upon by creating an inspection class map containing one or more **match** commands or you can use **match** commands directly in the inspection policy map. Some **match** commands let you identify text in a packet using a regular expression; for example, you can match URL strings inside HTTP packets. You can group regular expressions in a regular expression class map (see the **class-map type regex** command).

A regular expression matches text strings either literally as an exact string, or by using *metacharacters* so you can match multiple variants of a text string. You can use a regular expression to match the content of certain application traffic; for example, you can match body text inside an HTTP packet.

**Note** As an optimization, the security appliance searches on the deobfuscated URL. Deobfuscation compresses multiple forward slashes (/) into a single slash. For strings that commonly use double slashes, like "http://", be sure to search for "http:/" instead.

Table 23-1 lists the metacharacters that have special meanings.

***Table 23-1 regex Metacharacters***

| Character | Description | Notes |
|---|---|---|
| . | Dot | Matches any single character. For example, **d.g** matches dog, dag, dtg, and any word that contains those characters, such as doggonnit. |
| (*exp*) | Subexpression | A subexpression segregates characters from surrounding characters, so that you can use other metacharacters on the subexpression. For example, **d(o\|a)g** matches dog and dag, but **do\|ag** matches do and ag. A subexpression can also be used with repeat quantifiers to differentiate the characters meant for repetition. For example, **ab(xy){3}z** matches abxyxyxyz. |
| \| | Alternation | Matches either expression it separates. For example, **dog\|cat** matches dog or cat. |
| ? | Question mark | A quantifier that indicates that there are 0 or 1 of the previous expression. For example, **lo?se** matches lse or lose.<br>**Note** You must enter **Ctrl+V** and then the question mark or else the help function is invoked. |
| * | Asterisk | A quantifier that indicates that there are 0, 1 or any number of the previous expression. For example, **lo*se** matches lse, lose, loose, and so on. |
| + | Plus | A quantifier that indicates that there is at least 1 of the previous expression. For example, **lo+se** matches lose and loose, but not lse. |
| {*x*} or {*x*,} | Minimum repeat quantifier | Repeat at least *x* times. For example, **ab(xy){2,}z** matches abxyxyz, abxyxyxyz, and so on. |
| [*abc*] | Character class | Matches any character in the brackets. For example, **[abc]** matches a, b, or c. |
| [^*abc*] | Negated character class | Matches a single character that is not contained within the brackets. For example, **[^abc]** matches any character other than a, b, or c. **[^A-Z]** matches any single character that is not an uppercase letter. |
| [*a*-*c*] | Character range class | Matches any character in the range. **[a-z]** matches any lowercase letter. You can mix characters and ranges: **[abcq-z]** matches a, b, c, q, r, s, t, u, v, w, x, y, z, and so does [**a-cq-z]**.<br>The dash (-) character is literal only if it is the last or the first character within the brackets: **[abc-]** or **[-abc]**. |
| "" | Quotation marks | Preserves trailing or leading spaces in the string. For example, **" test"** preserves the leading space when it looks for a match. |
| ^ | Caret | Specifies the beginning of a line. |

*Table 23-1 regex Metacharacters (continued)*

| Character | Description | Notes |
|---|---|---|
| **** | Escape character | When used with a metacharacter, matches a literal character. For example, **\[** matches the left square bracket. |
| *char* | Character | When character is not a metacharacter, matches the literal character. |
| **\r** | Carriage return | Matches a carriage return 0x0d. |
| **\n** | Newline | Matches a new line 0x0a. |
| **\t** | Tab | Matches a tab 0x09. |
| **\f** | Formfeed | Matches a form feed 0x0c. |
| **\x***NN* | Escaped hexadecimal number | Matches an ASCII character using hexadecimal (exactly two digits). |
| *NNN* | Escaped octal number | Matches an ASCII character as octal (exactly three digits). For example, the character 040 represents a space. |

To test a regular expression to make sure it matches what you think it will match, enter the **test regex** command.

The regular expression performance impact is determined by two main factors:

- The length of text that needs to be searched for a regular expression match.

  The regular expression engine has only a small impact to the security appliance performance when the search length is small.

- The number of regular expression chained tables that need to be searched for a regular expression match.

#### How the Search Length Impacts Performance

When you configure a regular expression search, every byte of the searched text is usually examined against a regular expression database to find a match. The longer the searched text is, the longer the search time will be. Below is a performance test case which illustrates this phenomenon.

- An HTTP transaction includes one 300-byte long GET request and one 3250-byte long response.
- 445 regular expressions for URI search and 34 regular expressions for request body search.
- 55 regular expressions for response body search.

When a policy is configured to search the URI and the body in the HTTP GET request only, the throughput is:

- 420 mbps when the corresponding regular expression database is not searched.
- 413 mbps when the corresponding regular expression database is searched (this demonstrates a relatively small overhead of using regular expression).

But when a policy is configured to also search the whole HTTP response body, the throughput drops down to 145 mbps because of the long response body (3250 bytes) search.

Following is a list of factors that will increase the length of text for a regular expression search:

- A regular expression search is configured on multiple, different protocol fields. For example, in HTTP inspection, if only URI is configured for a regular expression match, then only the URI field is searched for a regular expression match, and the search length is then limited to the URI length. But if additional protocol fields are also configured for a regular expression match, such as Headers, Body, and so on, then the search length will increase to include the header length and body length.
- The field to be searched is long. For example, if the URI is configured for a regular expression search, then a long URI in a GET request will have a long search length. Also, currently the HTTP body search length is limited by default to 200 bytes. If, however, a policy is configured to search the body, and the body search length is changed to 5000 bytes, then there will be severe impact on the performance because of the long body search.

#### How the Number of Chained Regular Expression Tables Impact Performance

Currently, all regular expressions that are configured for the same protocol field, such as all regular expressions for URI, are built into a database consisting of one or more regular expression chained tables. The number of tables is determined by the total memory required and the availability of memory at the time the tables are built. A regular expression database will be split into multiple tables under any of the following conditions:

- When the total memory required is greater than 32 MB since the maximum table size is limited to 32 MB.
- When the size of the largest contiguous memory is not sufficient to build a complete regular expression database, then smaller but multiple tables will be built to accommodate all the regular expressions. Note that the degree of memory fragmentation varies depending on many factors that are interrelated and are almost impossible to predict the level of fragmentation.

With multiple chained tables, each table must be searched for regular expression matches and hence the search time increases in proportion to the number of tables that are searched.

Certain types of regular expressions tend to increase the table size significantly. It is prudent to design regular expressions in a way to avoid wildcard and repeating factors if possible. See Table 23-1 for a description of the following metacharacters:

- Regular expressions with wildcard type of specifications:
  - Dot (.)
- Various character classes that match any character in a class:
  - [^a-z]
  - [a-z]
  - [abc]]
- Regular expressions with repeating type of specifications:
  - *
  - +
  - {n,}
- Combination of the wild-card and repeating types of regular expressions can increase the table size dramatically, for examples:
  - 123.*xyz
  - 123.+xyz
  - [^a-z]+
  - [^a-z]*

  - .*123.* (This should not be done because this is equivalent to matching "123").

The following examples illustrate how memory consumptions are different for regular expressions with and without wildcards and repetition.

- Database size for the following 4 regular expressions is 958,464 bytes.

```
regex r1 "q3rfict9(af.*12)*ercvdf"
regex r2 "qtaefce.*qeraf.*adasdfev"
regex r3 "asdfdfdfds.*wererewr0e.*aaaxxxx.*xxx"
regex r4 "asdfdfdfds.*wererewr0e.*afdsvcvr.*aefdd"
```

- Database size for the following 4 regular expressions is only 10240 bytes.

```
regex s1 "abcde"
regex s2 "12345"
regex s3 "123xyz"
regex s4 "xyz123"
```

A large number of regular expressions will increase the total memory that is needed for the regular expression database and hence increases the probabilities of more tables if memory is fragmented. Following are examples of memory consumptions for different numbers of regular expressions:

- 100 sample URIs: 3,079,168 bytes
- 200 sample URIs: 7,156,224 bytes
- 500 sample URIs: 11,198,971 bytes

**Note** The maximum number of regular expressions per context is 2048.

The **debug menu regex 40 10** command can be used to display how many chained tables there are in each regex database.

## Examples

The following example creates two regular expressions for use in an inspection policy map:

```
hostname(config)# regex url_example example\.com
hostname(config)# regex url_example2 example2\.com
```

## Related Commands

| Command | Description |
|---|---|
| **class-map type inspect** | Creates ain inspection class map to match traffic specific to an application. |
| **policy-map** | Creates a policy map by associating the traffic class with one or more actions. |
| **policy-map type inspect** | Defines special actions for application inspection. |
| **class-map type regex** | Creates a regular expression class map. |
| **test regex** | Tests a regular expression. |

# reload

To reboot and reload the configuration, use the **reload** command in privileged EXEC mode.

**reload** [**at** *hh*:*mm* [*month day* | *day month*]] [**cancel**] [**in** [*hh*:]*mm*] [**max-hold-time** [*hh*:]*mm*] [**noconfirm**] [**quick**] [**reason** *text*] [**save-config**]

## Syntax Description

| | |
|---|---|
| **at** *hh*:*mm* | (Optional) Schedules a reload of the software to take place at the specified time (using a 24-hour clock). If you do not specify the month and day, the reload occurs at the specified time on the current day (if the specified time is later than the current time), or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight. The reload must take place within 24 hours. |
| **cancel** | (Optional) Cancels a scheduled reload. |
| *day* | (Optional) Number of the day in the range from 1 to 31. |
| **in** [*hh*:]*mm* | (Optional) Schedules a reload of the software to take effect in the specified minutes or hours and minutes. The reload must occur within 24 hours. |
| **max-hold-time** [*hh*:]*mm* | (Optional) Specifies the maximum hold time the security appliance waits to notify other subsystems before a shutdown or reboot. After this time elapses, a quick (forced) shutdown/reboot occurs. |
| *month* | (Optional) Specifies the name of the month. Enter enough characters to create a unique string for the name of the month. For example, "Ju" is not unique because it could represent June or July, but "Jul" is unique because no other month beginning with those exact three letters. |
| **noconfirm** | (Optional) Permits the security appliance to reload without user confirmation. |
| **quick** | (Optional) Forces a quick reload, without notifying or properly shutting down all the subsystems. |
| **reason** *text* | (Optional) Specifies the reason for the reload, 1 to 255 characters. The reason text is sent to all open IPSec VPN client, terminal, console, telnet, SSH, and ASDM connections/sessions.<br><br>**Note** Some applications, like isakmp, require additional configuration to send the reason text to IPSec VPN Clients. Refer to the appropriate section in the software configuration documentation for more information. |
| **save-config** | (Optional) Saves the running configuration to memory before shutting down. If you do not enter the **save-config** keyword, any configuration changes that have not been saved will be lost after the reload. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC | • | • | • | — | • |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was modified to add the following new arguments and keywords: *day*, *hh*, *mm*, *month*, **quick**, **save-config**, and *text*. |

## Usage Guidelines

The  command lets you reboot the security appliance and reload the configuration from Flash.

By default, the **reload** command is interactive. The security appliance first checks whether the configuration has been modified but not saved. If so, the security appliance prompts you to save the configuration. In multiple context mode, the security appliance prompts for each context with an unsaved configuration. If you specify the **save-config** parameter, the configuration is saved without prompting you. The security appliance then prompts you to confirm that you really want to reload the system. Only a response of **y** or pressing the **Enter** key causes a reload. Upon confirmation, the security appliance starts or schedules the reload process, depending upon whether you have specified a delay parameter (**in** or **at**).

By default, the reload process operates in "graceful" (also known as "nice") mode. All registered subsystems are notified when a reboot is about to occur, allowing these subsystems to shut down properly before the reboot. To avoid waiting until for such a shutdown to occur, specify the **max-hold-time** parameter to specify a maximum time to wait. Alternatively, you can use the **quick** parameter to force the reload process to begin abruptly, without notifying the affected subsystems or waiting for a graceful shutdown.

You can force the **reload** command to operate noninteractively by specifying the **noconfirm** parameter. In this case, the security appliance does not check for an unsaved configuration unless you have specified the **save-config** parameter. The security appliance does not prompt the user for confirmation before rebooting the system. It starts or schedules the reload process immediately, unless you have specified a delay parameter, although you can specify the **max-hold-time** or **quick** parameters to control the behavior or the reload process.

Use **reload cancel** to cancel a scheduled reload. You cannot cancel a reload that is already in progress.

**Note** Configuration changes that are not written to the Flash partition are lost after a reload. Before rebooting, enter the **write memory** command to store the current configuration in the Flash partition.

## Examples

This example shows how to reboot and reload the configuration:

```
hostname# reload
Proceed with ?  [confirm] y

Rebooting...

XXX Bios VX.X
...
```

**Related Commands**

| Command | Description |
|---|---|
| **show reload** | Displays the reload status of the security appliance. |

# remote-access threshold session-threshold-exceeded

To set threshold values, use the **remote-access threshold** command in global configuration mode. To remove threshold values, use the **no** version of this command. This command specifies the number of active remote access sessions, at which point the security appliance sends traps.

**remote-access threshold session-threshold-exceeded** {*threshold-value*}

no **remote-access threshold session-threshold-exceeded**

## Syntax Description

| | |
|---|---|
| *threshold-value* | Specifies an integer less than or equal to the session limit the security appliance supports. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | • | — | — | • |

## Command History

| Release | Modification |
|---|---|
| 7.0 (1) | This command was introduced. |

## Examples

The following example shows how to set a threshold value of 1500:

```
hostname# remote-access threshold session-threshold-exceeded 1500
```

## Related Commands

| Command | Description |
|---|---|
| **snmp-server enable trap remote-access** | Enables threshold trapping. |

# rename

To rename a file or a directory from the source filename to the destination filename, use the **rename** command in privileged EXEC mode.

**rename** ***[/noconfirm] [*****disk0: | disk1: | flash:*****] source-path*** **[disk0: | disk1: | flash:]** ***destination-path***

## Syntax Description

| | |
|---|---|
| **/noconfirm** | (Optional) Suppresses the confirmation prompt. |
| *destination-path* | Specifies the path of the destination file. |
| **disk0**: | (Optional) Specifies the internal Flash memory, followed by a colon. |
| **disk1**: | (Optional) Specifies the external Flash memory card, followed by a colon. |
| **flash:** | (Optional) Specifies the internal Flash memory, followed by a colon. |
| *source-path* | Specifies the path of the source file. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Privileged EXEC | • | • | • | — | • |

## Command History

| Release | Modification |
|---|---|
| 7.0 | This command was introduced. |

## Usage Guidelines

The **rename flash: flash:** command prompts you to enter a source and destination filename.

You cannot rename a file or directory across file systems.

For example:

```
hostname# rename flash: disk1:
Source filename []? new-config
Destination filename []? old-config
%Cannot rename between filesystems
```

## Examples

The following example shows how to rename a file named “test” to “test1”:

```
hostname# rename flash: flash:
Source filename [running-config]? test
Destination filename [n]? test1
```

## Related Commands

| Command | Description |
|---|---|
| **mkdir** | Creates a new directory. |
| **rmdir** | Removes a directory. |
| **show file** | Displays information about the file system. |

# rename (class-map)

To rename a class map, enter the **rename** command in class-map configuration mode.

**rename** *new_name*

## Syntax Description

| | |
|---|---|
| *new_name* | Specifies the new name of the class map, up to 40 characters in length. The name "class-default" is reserved. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Class-map configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Examples

The following example shows how to rename a class map from test to test2:

```
hostname(config)# class-map test
hostname(config-cmap)# rename test2
```

## Related Commands

| Command | Description |
|---|---|
| **class-map** | Creates a class map. |

# renewal-reminder

To specify the number of days prior to local Certificate Authority (CA) certificate expiration that an initial reminder to re-enroll is sent to certificate owners, use the **renewal-reminder** command in CA server configuration mode. To reset the time to the default of 14 days, use the **no** form of this command.

**renewal-reminder** *time*

**no renewal-reminder**

## Syntax Description

| | |
|---|---|
| *time* | Specifies the time in days prior to the expiration of an issued certificate that the certificate owner is first reminded to re-enroll. Valid values range from 1 to 90 days. |

## Defaults

By default, the CA server sends an expiration notice and reminder to re-enroll 14 days prior to certificate expiration.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| CA server configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

There are three reminders in all: one at the renewal-reminder time prior to certificate expiration, a second at (expiration time + otp expiration) - renewal-reminder time/2, and a third at (expiration time + otp expiration) - renewal-reminder/4.

An e-mail is sent automatically to the certificate owner for each of the three reminders, if an e-mail address is specified in the user database. If no e-mail address exists, a syslog message is generated to alert the administrator of the renewal.

## Examples

The following example specifies that the security appliance send an expiration notice to users 7 days prior to certificate expiration:

```
hostname(config)# crypto ca server
hostname(config-ca-server)# renewal-reminder 7
hostname(config-ca-server)#
```

The following example resets the expiration notice time to the default of 14 days prior to certificate expiration:

```
hostname(config)# crypto ca server
hostname(config-ca-server)# no renewal-reminder
hostname(config-ca-server)#
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ca server** | Provides access to the CA Server Configuration mode CLI command set, which allows you to configure and manage the local CA. |
| **lifetime** | Specifies the lifetimes of the CA certificate, all issued certificates, and the CRL. |
| **show crypto ca server** | Displays the configuration details of the local CA server. |

# replication http

To enable HTTP connection replication for the failover group, use the **replication http** command in failover group configuration mode. To disable HTTP connection replication, use the **no** form of this command.

**replication http**

**no replication http**

## Syntax Description

This command has no arguments or keywords.

## Defaults

Disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Failover group configuration | • | • | — | — | • |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Usage Guidelines

By default, the security appliance does not replicate HTTP session information when Stateful Failover is enabled. Because HTTP sessions are typically short-lived, and because HTTP clients typically retry failed connection attempts, not replicating HTTP sessions increases system performance without causing serious data or connection loss. The **replication http** command enables the stateful replication of HTTP sessions in a Stateful Failover environment, but could have a negative effect on system performance.

This command is available for Active/Active failover only. It provides the same functionality as the **failover replication http** command for Active/Standby failover, except for failover groups in Active/Active failover configurations.

## Examples

The following example shows a possible configuration for a failover group:

```
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# replication http
hostname(config-fover-group)# exit
```

**Related Commands**

| Command | Description |
|---|---|
| **failover group** | Defines a failover group for Active/Active failover. |
| **failover replication http** | Configures stateful failover to replicate HTTP connections. |

# request-command deny

To disallow specific commands within FTP requests, use the **request-command deny** command in FTP map configuration mode, which is accessible by using the **ftp-map** command. To remove the configuration, use the **no** form of this command.

**request-command deny** { **appe** | **cdup** | **dele** | **get** | **help** | **mkd** | **put** | **rmd** | **rnfr** | **rnto** | **site** | **stou** }

**no request-command deny** { **appe** | **cdup** | **help** | **retr** | **rnfr** | **rnto** | **site** | **stor** | **stou** }

## Syntax Description

| | |
|---|---|
| **appe** | Disallows the command that appends to a file. |
| **cdup** | Disallows the command that changes to the parent directory of the current working directory. |
| **dele** | Disallows the command that deletes a file on the server. |
| **get** | Disallows the client command for retrieving a file from the server. |
| **help** | Disallows the command that provides help information. |
| **mkd** | Disallows the command that makes a directory on the server. |
| **put** | Disallows the client command for sending a file to the server. |
| **rmd** | Disallows the command that deletes a directory on the server. |
| **rnfr** | Disallows the command that specifies rename-from filename. |
| **rnto** | Disallows the command that specifies rename-to filename. |
| **site** | Disallows the command that is specific to the server system. Usually used for remote administration. |
| **stou** | Disallows the command that stores a file using a unique file name. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| FTP map configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Usage Guidelines

This command is used for controlling the commands allowed within FTP requests traversing the security appliance when using strict FTP inspection.

## Examples

The following example causes the security appliance to drop FTP requests containing **stor**, **stou**, or **appe** commands:

```
hostname(config)# ftp-map inbound_ftp
hostname(config-ftp-map)# request-command deny put stou appe
```

## Related Commands

| Commands | Description |
|---|---|
| **class-map** | Defines the traffic class to which to apply security actions. |
| **ftp-map** | Defines an FTP map and enables FTP map configuration mode. |
| **inspect ftp** | Applies a specific FTP map to use for application inspection. |
| **mask-syst-reply** | Hides the FTP server response from clients. |
| **policy-map** | Associates a class map with specific security actions. |

# request-data-size

To set the size of the payload in the SLA operation request packets, use the **request-data-size** command in SLA monitor protocol configuration mode. To restore the default value, use the **no** form of this command.

**request-data-size** *bytes*

**no request-data-size**

## Syntax Description

| | |
|---|---|
| *bytes* | The size, in bytes, of the request packet payload. Valid values are from 0 to 16384. The minimum value depends upon the protocol used. For echo types, the minimum value is 28 bytes. Do not set this value higher than the maximum allowed by the protocol or the PMTU.<br>**Note** The security appliance adds an 8 byte timestamp to the payload, so the actual payload is *bytes* + 8. |

## Defaults

The default *bytes* is 28.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| SLA monitor protocol configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

For reachability, it may be necessary to increase the default data size to detect PMTU changes between the source and the target. Low PMTU will likely affect session performance and, if detected, may indicate that the secondary path be used.

## Examples

The following example configures an SLA operation with an ID of 123 that uses an ICMP echo request/response time probe operation. It sets the payload size of the echo request packets to 48 bytes and the number of echo requests sent during an SLA operation to 5.

```
hostname(config)# sla monitor 123
hostname(config-sla-monitor)# type echo protocol ipIcmpEcho 10.1.1.1 interface outside
hostname(config-sla-monitor-echo)# num-packets 5
hostname(config-sla-monitor-echo)# request-data-size 48
hostname(config-sla-monitor-echo)# timeout 4000
```

```
hostname(config-sla-monitor-echo)# threshold 2500
hostname(config-sla-monitor-echo)# frequency 10
hostname(config)# sla monitor schedule 123 life forever start-time now
hostname(config)# track 1 rtr 123 reachability
```

## Related Commands

| Command | Description |
|---|---|
| **num-packets** | Specifies the number of request packets to send during an SLA operation. |
| **sla monitor** | Defines an SLA monitoring operation. |
| **type echo** | Configures the SLA operation as an echo response time probe operation. |

# request-queue

To specify the maximum number of GTP requests that will be queued waiting for a response, use the **request-queue** command in GTP map configuration mode, which is accessed by using the **gtp-map** command. Use the **no** form of this command to return this number to the default of 200.

**request-queue** *max_requests*

**no request-queue** *max_requests*

## Syntax Description

| | |
|---|---|
| *max_requests* | The maximum number of GTP requests that will be queued waiting for a response. The range values is 1 to 4294967295. |

## Defaults

The *max_requests* default is 200.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| GTP map configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Usage Guidelines

The **gtp request-queue** command specifies the maximum number of GTP requests that are queued waiting for a response. When the limit has been reached and a new request arrives, the request that has been in the queue for the longest time is removed. The Error Indication, the Version Not Supported and the SGSN Context Acknowledge messages are not considered as requests and do not enter the request queue to wait for a response.

## Examples

The following example specifies a maximum request queue size of 300 bytes:

```
hostname(config)# gtp-map qtp-policy
hostname(config-gtpmap)# request-queue-size 300
```

## Related Commands

| Commands | Description |
|---|---|
| **clear service-policy inspect gtp** | Clears global GTP statistics. |
| **debug gtp** | Displays detailed information about GTP inspection. |

| Commands | Description |
|---|---|
| **gtp-map** | Defines a GTP map and enables GTP map configuration mode. |
| **inspect gtp** | Applies a specific GTP map to use for application inspection. |
| **show service-policy inspect gtp** | Displays the GTP configuration. |

# request-timeout

To configure the number of seconds before a failed SSO authentication attempt times out, use the **request-timeout** command in webvpn configuration mode.

To return to the default value, use the **no** form of this command.

**request-timeout** *seconds*

**no request-timeout**

## Syntax Description

| | |
|---|---|
| *seconds* | The number of seconds before a failed SSO authentication attempt times out. The range is 1 to 30 seconds. Fractions are not supported. |

## Defaults

The default value for this command is 5 seconds.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Webvpn configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.1.1 | This command was introduced. |

## Usage Guidelines

Single sign-on support, available only for WebVPN, lets users access different secure services on different servers without entering a username and password more than once. The security appliance currently supports SiteMinder and SAML POST type SSO servers.

This command applies to both types of SSO Servers.

Once you have configured the security appliance to support SSO authentication, you have the option to adjust two timeout parameters:

- The number of seconds before a failed SSO authentication attempt times out using the **request-timeout** command.
- The number of times the security appliance retries a failed SSO authentication attempt. (See the **max-retry-attempts** command.)

## Examples

The following example, entered in webvpn-config-sso-siteminder mode, configures an authentication timeout at ten seconds for the SiteMinder type SSO server, “example”:

```
hostname(config-webvpn)# sso-server example type siteminder
hostname(config-webvpn-sso-siteminder)# request-timeout 10
```

## Related Commands

| Command | Description |
|---|---|
| **max-retry-attempts** | Configures the number of times the security appliance retries a failed SSO authentication attempt. |
| **policy-server-secret** | Creates a secret key used to encrypt authentication requests to a SiteMinder SSO server. |
| **show webvpn sso-server** | Displays the operating statistics for all SSO servers configured on the security device. |
| **sso-server** | Creates a single sign-on server. |
| **test sso-server** | Tests an SSO server with a trial authentication request. |
| **web-agent-url** | Specifies the SSO server URL to which the security appliance makes SiteMinder SSO authentication requests. |

# reserve-port-protect

To restrict usage on the reserve port during media negotiation, use the **reserve-port-protect** command in parameters configuration mode. Parameters configuration mode is accessible from policy map configuration mode. To disable this feature, use the **no** form of this command.

**reserve-port-protect**

**no reserve-port-protect**

## Syntax Description

This command has no arguments or keywords.

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Parameters configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Examples

The following example shows how to protect the reserve port in an RTSP inspection policy map:

```
hostname(config)# policy-map type inspect rtsp rtsp_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# reserve-port-protect
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Identifies a class map name in the policy map. |
| **class-map type inspect** | Creates an inspection class map to match traffic specific to an application. |
| **policy-map** | Creates a Layer 3/4 policy map. |
| **show running-config policy-map** | Display all current policy map configurations. |

# reserved-bits

To clear reserved bits in the TCP header, or drop packets with reserved bits set, use the **reserved-bits** command in tcp-map configuration mode. To remove this specification, use the **no** form of this command.

**reserved-bits** {**allow** | **clear** | **drop**}

**no reserved-bits** {**allow** | **clear** | **drop**}

## Syntax Description

| | |
|---|---|
| **allow** | Allows packet with the reserved bits in the TCP header. |
| **clear** | Clears the reserved bits in the TCP header and allows the packet. |
| **drop** | Drops the packet with the reserved bits in the TCP header. |

## Defaults

The reserved bits are allowed by default.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Tcp-map configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was introduced. |

## Usage Guidelines

The **tcp-map** command is used along with the Modular Policy Framework infrastructure. Define the class of traffic using the **class-map** command and customize the TCP inspection with **tcp-map** commands. Apply the new TCP map using the **policy-map** command. Activate TCP inspection with **service-policy** commands.

Use the **tcp-map** command to enter tcp-map configuration mode. Use the **reserved-bits** command in tcp-map configuration mode to remove ambiguity as to how packets with reserved bits are handled by the end host, which may lead to desynchronizing the security appliance. You can choose to clear the reserved bits in the TCP header or even drop packets with the reserved bits set.

## Examples

The following example shows how to clear packets on all TCP flows with the reserved bit set:

```
hostname(config)# access-list TCP extended permit tcp any any
hostname(config)# tcp-map tmap
hostname(config-tcp-map)# reserved-bits clear
hostname(config)# class-map cmap
hostname(config-cmap)# match access-list TCP
```

```
hostname(config)# policy-map pmap
hostname(config-pmap)# class cmap
hostname(config-pmap)# set connection advanced-options tmap
hostname(config)# service-policy pmap global
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Specifies a class map to use for traffic classification. |
| **policy-map** | Configures a policy; that is, an association of a traffic class and one or more actions. |
| **set connection** | Configures connection values. |
| **tcp-map** | Creates a TCP map and allows access to tcp-map configuration mode. |

# reset

When using the Modular Policy Framework, drop packets, close the connection, and send a TCP reset for traffic that matches a **match** command or class map by using the **reset** command in match or class configuration mode. This reset action is available in an inspection policy map (the **policy-map type inspect** command) for application traffic; however, not all applications allow this action. To disable this action, use the **no** form of this command.

**reset** [**log**]

**no reset** [**log**]

## Syntax Description

| | |
|---|---|
| **log** | Logs the match. The system log message number depends on the application. |

## Defaults

No default behaviors or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Match and class configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

An inspection policy map consists of one or more **match** and **class** commands. The exact commands available for an inspection policy map depends on the application. After you enter the **match** or **class** command to identify application traffic (the **class** command refers to an existing **class-map type inspect** command that in turn includes **match** commands), you can enter the **reset** command to drop packets and close the connection for traffic that matches the **match** command or **class** command.

If you reset a connection, then no further actions are performed in the inspection policy map. For example, if the first action is to reset the connection, then it will never match any further **match** or **class** commands. If the first action is to log the packet, then a second action, such as resetting the connection, can occur. You can configure both the **reset** and the **log** action for the same **match** or **class** command, in which case the packet is logged before it is reset for a given match.

When you enable application inspection using the **inspect** command in a Layer 3/4 policy map (the **policy-map** command), you can enable the inspection policy map that contains this action, for example, enter the **inspect http http_policy_map** command where http_policy_map is the name of the inspection policy map.

## Examples

The following example resets the connection and sends a log when they match the http-traffic class map. If the same packet also matches the second **match** command, it will not be processed because it was already dropped.

```
hostname(config-cmap)# policy-map type inspect http http-map1
hostname(config-pmap)# class http-traffic
hostname(config-pmap-c)# reset log
hostname(config-pmap-c)# match req-resp content-type mismatch
hostname(config-pmap-c)# reset log
```

## Related Commands

| Commands | Description |
|---|---|
| **class** | Identifies a class map name in the policy map. |
| **class-map type inspect** | Creates an inspection class map to match traffic specific to an application. |
| **policy-map** | Creates a Layer 3/4 policy map. |
| **policy-map type inspect** | Defines special actions for application inspection. |
| **show running-config policy-map** | Display all current policy map configurations. |

# retries

To specify the number of times to retry the list of DNS servers when the security appliance does not receive a response, use the **dns retries** command in global configuration mode. To restore the default setting, use the **no** form of this command.

**retries** *number*

**no retries** [*number*]

## Syntax Description

| | |
|---|---|
| *number* | Specifies the number of retries, from 0 through 10. The default is 2. |

## Defaults

The default number of retries is 2.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.1(1) | This command was introduced. |

## Usage Guidelines

Add DNS servers using the **name-server** command.

This command replaces the **dns name-server** command.

## Examples

The following example sets the number of retries to 0. The security appliance tries each server only once.

```
hostname(config)# dns server-group dnsgroup1
hostname(config-dns-server-group)# dns retries 0
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure dns** | Removes all DNS commands. |
| **dns server-group** | Enters the dns server-group mode. |
| **show running-config dns server-group** | Shows one or all the existing dns-server-group configurations. |

# retry-interval

To configure the amount of time between retry attempts for a particular AAA server designated in a prior aaa-server host command, use the **retry-interval** command in AAA-server host mode. To reset the retry interval to the default value, use the **no** form of this command.

**retry-interval** *seconds*

**no retry-interval**

## Syntax Description

| | |
|---|---|
| *seconds* | Specify the retry interval (1-10 seconds) for the request. This is the time the security appliance waits before retrying a connection request. |

## Defaults

The default retry interval is 10 seconds.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| AAA-server host | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was modified to conform to CLI guidelines. |

## Usage Guidelines

Use the **retry-interval** command to specify or reset the number of seconds the security appliance waits between connection attempts. Use the **timeout** command to specify the length of time during which the security appliance attempts to make a connection to a AAA server.

## Examples

The following examples show the **retry-interval** command in context.

```
hostname(config)# aaa-server svrgrp1 protocol radius
hostname(config-aaa-server-group)# aaa-server svrgrp1 host 1.2.3.4
hostname(config-aaa-server-host)# timeout 7
hostname(config-aaa-server-host)# retry-interval 9
hostname(config-aaa-server-host)#
```

## Related Commands

| Command | Description |
|---|---|
| **aaa-server host** | Enters AAA server host configuration mode so you can configure AAA server parameters that are host-specific. |

| | |
|---|---|
| **clear configure aaa-server** | Removes all AAA command statements from the configuration. |
| **show running-config aaa-server** | Displays AAA server statistics for all AAA servers, for a particular server group, for a particular server within a particular group, or for a particular protocol |
| **timeout** | Specifies the length of time during which the security appliance attempts to make a connection to a AAA server. |

# reval-period

To specify the interval between each successful posture validation in a NAC Framework session, use the **reval-period** command in nac-policy-nac-framework configuration mode. To remove the command from the NAC Framework policy, use the **no** form of this command.

**reval-period** *seconds*

**no reval-period** [*seconds*]

## Syntax Description

| | |
|---|---|
| *seconds* | Number of seconds between each successful posture validation. The range is 300 to 86400. |

## Defaults

The default value is 36000.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| nac-policy-nac-framework configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.3(0) | "nac-" removed from command name. Command moved from group-policy configuration mode to nac-policy-nac-framework configuration mode. |
| 7.2(1) | This command was introduced. |

## Usage Guidelines

The security appliance starts the revalidation timer after each successful posture validation. The expiration of this timer triggers the next unconditional posture validation. The security appliance maintains posture validation during revalidation. The default group policy becomes effective if the Access Control Server is unavailable during posture validation or revalidation.

## Examples

The following example changes the revalidation timer to 86400 seconds:

```
hostname(config-nac-policy-nac-framework)# reval-period 86400
hostname(config-nac-policy-nac-framework)
```

The following example removes the revalidation timer from the NAC policy:

```
hostname(config-nac-policy-nac-framework)# no reval-period
hostname(config-nac-policy-nac-framework)
```

## Related Commands

| Command | Description |
|---|---|
| **eou timeout** | Changes the number of seconds to wait after sending an EAP over UDP message to the remote host in a NAC Framework configuration. |
| **sq-period** | Specifies the interval between each successful posture validation in a NAC Framework session and the next query for changes in the host posture. |
| **nac-policy** | Creates and accesses a Cisco NAC policy, and specifies its type. |
| **debug nac** | Enables logging of NAC Framework events. |
| **eou revalidate** | Forces immediate posture revalidation of one or more NAC Framework sessions. |

# revert webvpn all

To remove all web-related data (customization, plug-in, translation table, URL list, and web content) from the security appliance flash memory, enter the **revert webvpn all** command in privileged EXEC mode.

**revert webvpn all**

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Use the **revert webvpn all** command to disable and remove all web-related information (customization, plug-in, translation table, URL list, and web content) from the flash memory of the security appliance. Removal of all web-related data returns default settings when applicable.

## Examples

The following command removes all of the web-related configuration data from the security appliance:

```
hostname# revert webvpn all
hostname
```

## Related Commands

| Command | Description |
|---|---|
| **show import webvpn** (*option*) | Displays various imported WebVPN data and plug-ins. currently present in flash memory on the security appliance. |

# revert webvpn customization

To remove a customization object from the security appliance cache memory, enter the **revert webvpn customization** command in privileged EXEC mode.

**revert webvpn customization** *name*

## Syntax Description

| | |
|---|---|
| *name* | Specifies the name of the customization object to be deleted. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Use the **revert webvpn customization** command to remove Clientless SSL VPN support for the specified customization and to remove it from the cache memory on the security appliance. Removal of a customization object returns default settings when applicable. A customization object contains the configuration parameters for a specific, named portal page.

Version 8.0 software extends the functionality for configuring customization, and the new process is incompatible with previous versions. During the upgrade to 8.0 software, the security appliance preserves a current configuration by using old settings to generate new customization objects. This process occurs only once, and is more than a simple transformation from the old format to the new one because the old values are only a partial subset of the new ones.

**Note** Version 7.2 portal customizations and URL lists work in the Beta 8.0 configuration only if clientless SSL VPN (WebVPN) is enabled on the appropriate interface in the Version 7.2(x) configuration file before you upgrade to Version 8.0.

## Examples

The following command removes the customization object named GroupB:

```
hostname# revert webvpn customization groupb
hostname
```

## Related Commands

| Command | Description |
|---|---|
| `customization` | Specifies the customization object to use for a tunnel-group, group, or user. |
| `export customization` | Exports a customization object. |
| `import customization` | Installs a customization object. |
| `revert webvpn all` | Removes all webvpn-related data (customization, plug-in, translation table, URL list, and web content). |
| `show webvpn customization` | Displays the current customization objects present on the flash device of the security appliance. |

# revert webvpn plug-in protocol

To remove a plug-in from the flash device of the security appliance, enter the **revert webvpn plug-in protocol** command in privileged EXEC mode.

**revert plug-in protocol** *protocol*

## Syntax Description

| | |
|---|---|
| *protocol* | Enter one of the following strings:<br>• **rdp**<br>The Remote Desktop Protocol plug-in lets the remote user connect to a computer running Microsoft Terminal Services.<br>• **ssh**<br>The Secure Shell plug-in lets the remote user establish a secure channel to a remote computer, or lets the remote user use Telnet to connect to a remote computer.<br>• **vnc**<br>The Virtual Network Computing plug-in lets the remote user use a monitor, keyboard, and mouse to view and control a computer with remote desktop sharing turned on. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Use the **revert webvpn plug-in protocol** command to disable and remove Clientless SSL VPN support for the specified Java-based client application, as well as to remove it from the flash drive of the security appliance.

## Examples

The following command removes support for RDP:

```
hostname# revert webvpn plug-in protocol rdp
hostname
```

## Related Commands

| Command | Description |
|---|---|
| `import webvpn plug-in protocol` | Copies the specified plug-in from a URL to the flash device of the security appliance. Clientless SSL VPN automatically supports the use of the Java-based client application for future sessions when you issue this command. |
| `show import webvpn plug-in` | Lists the plug-ins present on the flash device of the security appliance. |

# revert webvpn translation-table

To remove a translation table from the security appliance flash memory, enter the **revert webvpn translation-table** command in privileged EXEC mode.

**revert webvpn translation-table** *translationdomain language*

## Syntax Description

| | |
|---|---|
| *translationdomain* | Available translation domains:<br>• AnyConnect<br>• PortForwarder<br>• Banners<br>• CSD<br>• Customization<br>• URL List<br>• (Translations of messages from RDP, SSH, and VNC plug-ins.) |
| *language* | Specifies the character-encoding method to be deleted. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Use the **revert webvpn translation-table** command to disable and remove an imported translation table and to remove it from the flash memory on the security appliance. Removal of a translation table returns default settings when applicable.

## Examples

The following command removes the AnyConnect translation table, Dutch:

```
hostname# revert webvpn translation-table anyconnect dutch
hostname
```

## Related Commands

| Command | Description |
|---|---|
| `revert webvpn all` | Removes all webvpn-related data (customization, plug-in, translation table, URL-list, and web content) . |
| `show webvpn translation-table` | Displays the current translation tables currently present on the flash device of the security appliance. |

# revert webvpn url-list

To remove a URL list from the security appliance, enter the **revert webvpn url-list** command in privileged EXEC mode.

**revert webvpn url-list template** *name*

## Syntax Description

| | |
|---|---|
| **template** *name* | Specifies the name of a URL list. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Privileged EXEC mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Use the **revert webvpn url-list** command to disable and remove a current URL list from the flash drive of the security appliance. Removal of a url-list returns default settings when applicable.

The template argument used with the **revert webvpn url-list** command specifies the name of a previously configured list of URLs. To configure such a list, use the **url-list** command in global configuration mode.

## Examples

The following command removes the URL list, servers2:

```
hostname# revert webvpn url-list servers2
hostname
```

## Related Commands

| Command | Description |
|---|---|
| **revert webvpn all** | Removes all webvpn-related data (customization, plug-in, translation table, URL list, and web content) . |
| **show running-configuration url-list** | Displays the current set of configured URL list commands. |
| **url-list (WebVPN mode)** | Applies a list of WebVPN servers and URLs to a particular user or group policy. |

# revert webvpn webcontent

To remove a specified web object from a location in the security appliance flash memory, enter the **revert webvpn webcontent** command in privileged EXEC mode.

**revert webvpn webcontent** *filename*

## Syntax Description

| | |
|---|---|
| *filename* | Specifies the name of the flash memory file with the web content to be deleted. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Privileged EXEC mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

Use the **revert webvpn content** command to disable and remove a file containing the web content and to remove it from the flash memory of the security appliance. Removal of web content returns default settings when applicable.

## Examples

The following command removes the web content file, ABCLogo, from the security appliance flash memory:

```
hostname# revert webvpn webcontent abclogo
hostname
```

## Related Commands

| Command | Description |
|---|---|
| **revert webvpn all** | Removes all webvpn-related data (customization, plug-in, translation table, URL list, and web content). |
| **show webvpn webcontent** | Displays the web content currently present in flash memory on the security appliance. |

# revocation-check

To set one or more methods for revocation checking, use the **revocation-check** command in crypto ca trustpoint mode. The security appliance tries the methods in the order that you configure them, trying the second and third methods only if the previous method returns an error (for example, server down), as opposed to finding the status as revoked.

You can set a revocation checking method in the client certificate validating trustpoint and also configure no revocation checking (**revocation-check none)** in the responder certificate validating trustpoint. The **match certificate** command documentation includes step-by-step configuration example.

To restore the default revocation checking method, which is *none*, use the **no** version of this command.

**revocation-check** {[**crl**] [**none**] [**ocsp**]}

**no revocation-check**

## Syntax Description

| | |
|---|---|
| **crl** | Specifies that the security appliance should use CRL as the revocation checking method. |
| **none** | Specifies that the security appliance should interpret the certificate status as valid, even if all methods return an error. |
| **ocsp** | Specifies that the security appliance should use OCSP as the revocation checking method. |

## Defaults

The default value is *none*.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| crypto ca trustpoint mode | • | • | • | • | • |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. The following permutations replace previous commands:<br>• **revocation-check crl none** replaces **crl optional**<br>• **revocation-check crl** replaces **crl required**<br>• **revocation-check none** replaces **crl nocheck** |

## Usage Guidelines

The signer of the OCSP response is usually the OCSP server (responder) certificate. After receiving the response, devices try to verify the responder certificate.

Normally a CA sets the lifetime of its OCSP responder certificate to a relatively short period to minimize the chance of compromising its security. The CA includes an ocsp-no-check extension in the responder certificate that indicates it does not need revocation status checking. But if this extension is not present, the device tries to check the certificate's revocation status using the revocation methods you configure for the trustpoint with this **revocation-check** command.The OCSP responder certificate must be verifiable if it does not have an ocsp-no-check extension since the OCSP revocation check fails unless you also set the *none* option to ignore the status check.

## Examples

The following example shows how to set revocation methods of OCSP and CRL, in that order, for the trustpoint called newtrust.

```
hostname(config)# crypto ca trustpoint newtrust
hostname(config-ca-trustpoint)# revocation-check ocsp crl
hostname(config-ca-trustpoint)#
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ca trustpoint** | Enters crypto ca trustpoint mode. Use this command in global configuration mode. |
| **match certificate** | Configures an OCSP override rule, |
| **ocsp disable-nonce** | Disables the nonce extension of the OCSP request. |
| **ocsp url** | Specifies the OCSP server to use to check all certificates associated with a trustpoint. |

# rewrite

To disable content rewriting a particular application or type of traffic over a WebVPN connection, use the **rewrite** command in webvpn mode. To eliminate a rewrite rule, use the **no** form of this command with the rule number, which uniquely identifies the rule. To eliminate all rewriting rules, use the **no** form of the command without the rule number.

By default, the security appliance rewrites, or transforms, all WebVPN traffic.

**rewrite order** *integer* **{enable | disable} resource-mask** *string* **[name** *resource name***]**

**no rewrite order** *integer* **{enable | disable} resource-mask** *string* **[name** *resource name***]**

## Syntax Description

| | |
|---|---|
| **disable** | Defines this rewrite rule as a rule that disables content rewriting for the specified traffic. When you disable content rewriting, traffic does not go through the security appliance. |
| **enable** | Defines this rewrite rule as a rule that enables content rewriting for the specified traffic. |
| *integer* | Sets the order of the rule among all of the configured rules. The range is 1-65534. |
| **name** | (Optional) Identifies the name of the application or resource to which the rule applies. |
| **order** | Defines the order in which the security appliance applies the rule. |
| **resource-mask** | Identifies the application or resource for the rule. |
| *resource name* | (Optional) Specifies the application or resource to which the rule applies. Maximum 128 bytes. |
| *string* | Specifies the name of the application or resource to match that can contain a regular expression. You can use the following wildcards:<br>Specifies a pattern to match that can contain a regular expression. You can use the following wildcards:<br>* — Matches everything. You cannot use this wildcard by itself. It must accompany an alphanumeric string.<br>? —Matches any single character.<br>[!seq] — Matches any character not in sequence.<br>[seq] — Matches any character in sequence.<br>Maximum 300 bytes. |

## Defaults

The default is to rewrite everything.

## Command Modes

The following table shows the modes in which you can enter the command:

| Command Mode | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| | Routed | Transparent | Single | Context | System |
| Webvpn mode | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.1(1) | This command was introduced. |

## Usage Guidelines

The security appliance performs content rewriting for applications to insure that they render correctly over WebVPN connections. Some applications do not require this processing, such as external public websites. For these applications, you might choose to turn off content rewriting.

You can turn off content rewriting selectively by using the rewrite command with the disable option to let users browse specific sites directly without going through the security appliance. This is similar to split-tunneling in IPSec VPN connections.

You can use this command multiple times. The order in which you configure entries is important because the security appliance searches rewrite rules by order number and applies the first rule that matches.

## Examples

The following example shows how to configure a rewrite rule, order number of 1, that turns off content rewriting for URLS from cisco.com domains:

```
hostname(config-webpn)# rewrite order 2 disable resource-mask *cisco.com/*
```

## Related Commands

| Command | Description |
|---|---|
| **apcf** | Specifies nonstandard rules to use for a particular application. |
| **proxy-bypass** | Configures minimal content rewriting for a particular application. |

# re-xauth

To require that IPSec users reauthenticate on IKE rekey, issue the **re-xauth enable** command in group-policy configuration mode. To disable user reauthentication on IKE rekey, use the **re-xauth disable** command.

To remove the re-xauth attribute from the running configuration, use the **no** form of this command. This enables inheritance of a value for reauthentication on IKE rekey from another group policy.

**re-xauth** {**enable** [**extended**] | **disable**}

**no re-xauth**

## Syntax Description

| | |
|---|---|
| **disable** | Disables reauthentication on IKE rekey |
| **enable** | Enables reauthentication on IKE rekey |
| **extended** | Extends the time allowed for reentering authentication credentials until the maximum lifetime of the configured SA. |

## Defaults

Reauthentication on IKE rekey is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Group policy | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0.4 | The **extended** keyword was added. |
| 7.0(1) | This command was introduced. |

## Usage Guidelines

Reauthentication on IKE rekey applies only to IPSec connections.

If you enable reauthentication on IKE rekey, the security appliance prompts the user to enter a username and password during initial Phase 1 IKE negotiation and also prompts for user authentication whenever an IKE rekey occurs. Reauthentication provides additional security.

The user has 30 seconds to enter credentials, and up to three attempts before the SA expires at approximately two minutes and the tunnel terminates. Use the **extended** keyword to allow users to reenter authentication credentials until the maximum lifetime of the configured SA.

To check the configured rekey interval, in monitoring mode, issue the **show crypto ipsec sa** command to view the security association lifetime in seconds and lifetime in kilobytes of data.

**Note** The reauthentication fails if there is no user at the other end of the connection.

## Examples

The following example shows how to enable reauthentication on rekey for the group policy named FirstGroup:

```
hostname(config) #group-policy FirstGroup attributes
hostname(config-group-policy)# re-xauth enable
```

# rip send version

To specify the RIP version used to send RIP updates on an interface, use the **rip send version** command in interface configuration mode. To restore the defaults, use the **no** form of this command.

**rip send version** {[**1**] [**2**]}

**no rip send version**

## Syntax Description

| | |
|---|---|
| **1** | Specifies RIP Version 1. |
| **2** | Specifies RIP Version 2. |

## Defaults

The security appliance sends RIP Version 1 packets.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Interface configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

You can override the global RIP send version setting on a per-interface basis by entering the **rip send version** command on an interface.

If you specify RIP version 2, you can enable neighbor authentication and use MD5-based encryption to authenticate the RIP updates.

## Examples

The following example configures the security appliance to send and receive RIP Version 1 and 2 packets on the specified interface:

```
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# rip send version 1 2
hostname(config-if)# rip receive version 1 2
```

## Related Commands

| Command | Description |
|---|---|
| **rip receive version** | Specifies the RIP version to accept when receiving updates on a specific interface. |
| **router rip** | Enables the RIP routing process and enter router configuration mode for that process. |
| **version** | Specifies the version of RIP used globally by the security appliance. |

# rip receive version

To specify the version of RIP accepted on an interface, use the **rip receive version** command in interface configuration mode. To restore the defaults, use the **no** form of this command.

**version** {[**1**] [**2**]}

**no version**

## Syntax Description

| | |
|---|---|
| **1** | Specifies RIP Version 1. |
| **2** | Specifies RIP Version 2. |

## Defaults

The security appliance accepts Version 1 and Version 2 packets.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Interface configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

You can override the global setting on a per-interface basis by entering the **rip receive version** command on an interface.

If you specify RIP version 2, you can enable neighbor authentication and use MD5-based encryption to authenticate the RIP updates.

## Examples

The following example configures the security appliance to receive RIP Version 1 and 2 packets the specified interface:

```
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# rip send version 1 2
hostname(config-if)# rip receive version 1 2
```

## Related Commands

| Command | Description |
|---|---|
| **rip send version** | Specifies the RIP version to use when sending update out of a specific interface. |
| **router rip** | Enables the RIP routing process and enter router configuration mode for that process. |
| **version** | Specifies the version of RIP used globally by the security appliance. |

# rip authentication mode

To specify the type of authentication used in RIP Version 2 packets, use the **rip authentication mode** command in interface configuration mode. To restore the default authentication method, use the **no** form of this command.

**rip authentication mode** {**text** | **md5**}

**no rip authentication mode**

## Syntax Description

| | |
|---|---|
| **md5** | Uses MD5 for RIP message authentication. |
| **text** | Uses clear text for RIP message authentication (not recommended). |

## Defaults

Clear text authentication is used by default.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Interface configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

If you specify RIP version 2, you can enable neighbor authentication and use MD5-based encryption to authenticate the RIP updates.

Use the **show interface** command to view the **rip authentication** commands on an interface.

## Examples

The following examples shows RIP authentication configured on interface GigabitEthernet0/3:

```
hostname(config)# interface Gigabit0/3
hostname(config-if)# rip authentication mode md5
hostname(config-if)# rip authentication key thisismykey key_id 5
```

## Related Commands

| Command | Description |
|---|---|
| **rip authentication key** | Enables RIP Version 2 authentication and specifies the authentication key. |
| **rip receive version** | Specifies the RIP version to accept when receiving updates on a specific interface. |

| Command | Description |
|---|---|
| **rip send version** | Specifies the RIP version to use when sending update out of a specific interface. |
| **show running-config interface** | Displays the configuration commands for the specified interface. |
| **version** | Specifies the version of RIP used globally by the security appliance. |

# rip authentication key

To enable authentication of RIP Version 2 packets and specify the authentication key, use the **rip authentication key** command in interface configuration mode. To disable RIP Version 2 authentication, use the **no** form of this command.

**rip authentication key** *key* **key_id** *key_id*

**no rip authentication key**

## Syntax Description

| | |
|---|---|
| *key* | Key to authenticate RIP updates. The key can contain up to 16 characters. |
| *key_id* | Key identification value; valid values range from 1 to 255. |

## Defaults

RIP authentication is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Interface configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

If you specify RIP version 2, you can enable neighbor authentication and use MD5-based encryption to authenticate the RIP updates. When you enable neighbor authentication, you must ensure that the *key* and *key_id* arguments are the same as those used by neighbor devices that provide RIP version 2 updates. The *key* is a text string of up to 16 characters.

Use the **show interface** command to view the **rip authentication** commands on an interface.

## Examples

The following examples shows RIP authentication configured on interface GigabitEthernet0/3:

```
hostname(config)# interface Gigabit0/3
hostname(config-if)# rip authentication mode md5
hostname(config-if)# rip authentication key thisismykey key_id 5
```

## Related Commands

| Command | Description |
|---|---|
| **rip authentication mode** | Specifies the type of authentication used in RIP Version 2 packets. |
| **rip receive version** | Specifies the RIP version to accept when receiving updates on a specific interface. |
| **rip send version** | Specifies the RIP version to use when sending update out of a specific interface. |
| **show running-config interface** | Displays the configuration commands for the specified interface. |
| **version** | Specifies the version of RIP used globally by the security appliance. |

# rip receive version

To specify the version of RIP accepted on an interface, use the **rip receive version** command in interface configuration mode. To restore the defaults, use the **no** form of this command.

**version** {[**1**] [**2**]}

**no version**

## Syntax Description

| | |
|---|---|
| **1** | Specifies RIP Version 1. |
| **2** | Specifies RIP Version 2. |

## Defaults

The security appliance accepts Version 1 and Version 2 packets.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Interface configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

You can override the global setting on a per-interface basis by entering the **rip receive version** command on an interface.

If you specify RIP version 2, you can enable neighbor authentication and use MD5-based encryption to authenticate the RIP updates.

## Examples

The following example configures the security appliance to receive RIP Version 1 and 2 packets the specified interface:

```
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# rip send version 1 2
hostname(config-if)# rip receive version 1 2
```

## Related Commands

| Command | Description |
|---|---|
| **rip send version** | Specifies the RIP version to use when sending update out of a specific interface. |
| **router rip** | Enables the RIP routing process and enter router configuration mode for that process. |
| **version** | Specifies the version of RIP used globally by the security appliance. |

# rip send version

To specify the RIP version used to send RIP updates on an interface, use the **rip send version** command in interface configuration mode. To restore the defaults, use the **no** form of this command.

**rip send version** {[**1**] [**2**]}

**no rip send version**

## Syntax Description

| | |
|---|---|
| **1** | Specifies RIP Version 1. |
| **2** | Specifies RIP Version 2. |

## Defaults

The security appliance sends RIP Version 1 packets.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Interface configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

You can override the global RIP send version setting on a per-interface basis by entering the **rip send version** command on an interface.

If you specify RIP version 2, you can enable neighbor authentication and use MD5-based encryption to authenticate the RIP updates.

## Examples

The following example configures the security appliance to send and receive RIP Version 1 and 2 packets on the specified interface:

```
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# rip send version 1 2
hostname(config-if)# rip receive version 1 2
```

## Related Commands

| Command | Description |
|---|---|
| **rip receive version** | Specifies the RIP version to accept when receiving updates on a specific interface. |
| **router rip** | Enables the RIP routing process and enter router configuration mode for that process. |
| **version** | Specifies the version of RIP used globally by the security appliance. |

# rmdir

To remove the existing directory, use the **rmdir** command in privileged EXEC mode.

**rmdir** [**/noconfirm**] [**disk0: | disk1: | flash:**]*path*

## Syntax Description

| | |
|---|---|
| **noconfirm** | (Optional) Suppresses the confirmation prompt. |
| **disk0***:* | (Optional) Specifies the nonremovable internal Flash memory, followed by a colon. |
| **disk1**: | (Optional) Specifies the removable external Flash memory card, followed by a colon. |
| **flash**: | (Optional) Specifies the nonremovable internal Flash, followed by a colon. In the ASA 5500 series, the **flash** keyword is aliased to **disk0**. |
| *path* | (Optional) The absolute or relative path of the directory to remove. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Privileged EXEC | • | • | • | — | • |

## Command History

| Release | Modification |
|---|---|
| 7.0 | This command was introduced. |

## Usage Guidelines

If the directory is not empty, the **rmdir** command fails.

## Examples

This example shows how to remove an existing directory named “test”:

```
hostname# rmdir test
```

## Related Commands

| Command | Description |
|---|---|
| **dir** | Displays the directory contents. |
| **mkdir** | Creates a new directory. |
| **pwd** | Displays the current working directory. |
| **show file** | Displays information about the file system. |

# route

To enter a static or default route for the specified interface, use the **route** command in global configuration mode. Use the **no** form of this command to remove routes from the specified interface.

**route** *interface_name ip_address netmask gateway_ip* [[*metric*] [**track** *number*] | **tunneled**]

**no route** *interface_name ip_address netmask gateway_ip* [[*metric*] [**track** *number*] | **tunneled**]

## Syntax Description

| | |
|---|---|
| *gateway_ip* | Specifies the IP address of the gateway router (the next-hop address for this route).<br>**Note** The *gateway_ip* argument is optional in transparent mode. |
| *interface_name* | Internal or external network interface name through which the traffic is routed. |
| *ip_address* | Internal or external network IP address. |
| *metric* | (Optional) The administrative distance for this route. Valid values range from 1 to 255. The default value is 1. |
| *netmask* | Specifies a network mask to apply to *ip_address*. |
| **track** *number* | (Optional) Associates a tracking entry with this route. Valid values are from 1 to 500.<br>**Note** The **track** option is only available in single, routed mode. |
| **tunneled** | Specifies route as the default tunnel gateway for VPN traffic. |

## Defaults

The *metric* default is 1.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| Preexisting | This command was preexisting. |
| 7.2(1) | The **track** *number* value was added. |

## Usage Guidelines

Use the **route** command to enter a default or static route for an interface. To enter a default route, set *ip_address* and *netmask* to **0.0.0.0,** or use the shortened form of **0**. All routes that are entered using the **route** command are stored in the configuration when it is saved.

You can define a separate default route for tunneled traffic along with the standard default route. When you create a default route with the **tunneled** option, all traffic from a tunnel terminating on the security appliance that cannot be routed using learned or static routes, is sent to this route. For traffic emerging from a tunnel, this route overrides over any other configured or learned default routes.

The following restrictions apply to default routes with the **tunneled** option:

- Do not enable unicast RPF (**ip verify reverse-path**) on the egress interface of tunneled route. Enabling uRPF on the egress interface of a tunneled route causes the session to fail.
- Do not enable TCP intercept on the egress interface of the tunneled route. Doing so causes the session to fail.
- Do not use the VoIP inspection engines (CTIQBE, H.323, GTP, MGCP, RTSP, SIP, SKINNY), the DNS inspect engine, or the DCE RPC inspection engine with tunneled routes. These inspection engines ignore the tunneled route.

You cannot define more than one default route with the **tunneled** option; ECMP for tunneled traffic is not supported.

Create static routes to access networks that are connected outside a router on any interface. For example, the security appliance sends all packets that are destined to the 192.168.42.0 network through the 192.168.1.5 router with this static **route** command.

```
hostname(config)# route dmz 192.168.42.0 255.255.255.0 192.168.1.5 1
```

Once you enter the IP address for each interface, the security appliance creates a CONNECT route in the route table. This entry is not deleted when you use the **clear route** or **clear configure route** commands.

If the **route** command uses the IP address from one of the interfaces on the security appliance as the gateway IP address, the security appliance will ARP for the destination IP address in the packet instead of ARPing for the gateway IP address.

## Examples

The following example shows how to specify one default **route** command for an outside interface:

```
hostname(config)# route outside 0 0 209.165.201.1 1
```

The following example shows how to add these static **route** commands to provide access to the networks:

```
hostname(config)# route dmz1 10.1.2.0 255.0.0.0 10.1.1.4 1
hostname(config)# route dmz1 10.1.3.0 255.0.0.0 10.1.1.4 1
```

The following example uses an SLA operation to install a default route to the 10.1.1.1 gateway on the outside interface. The SLA operation monitors the availability of that gateway.If the SLA operation fails, then the backup route on the dmz interface is used.

```
hostname(config)# sla monitor 123
hostname(config-sla-monitor)# type echo protocol ipIcmpEcho 10.1.1.1 interface outside
hostname(config-sla-monitor-echo)# timeout 1000
hostname(config-sla-monitor-echo)# frequency 3
hostname(config)# sla monitor schedule 123 life forever start-time now
hostname(config)# track 1 rtr 123 reachability
hostname(config)# route outside 0.0.0.0 0.0.0.0 10.1.1.1 track 1
hostname(config)# route dmz 0.0.0.0 0.0.0.0 10.2.1.1 254
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure route** | Removes statically configured **route** commands. |
| **clear route** | Removes routes learned through dynamic routing protocols such as RIP. |
| **show route** | Displays route information. |
| **show running-config route** | Displays configured routes. |

# route-map

To define the conditions for redistributing routes from one routing protocol into another, use the **route-map** command in global configuration mode. To delete a map, use the **no** form of this command.

**route-map** *map_tag* [**permit** | **deny**] [*seq_num*]

**no route-map** *map_tag* [**permit** | **deny**] [*seq_num*]

## Syntax Description

| | |
|---|---|
| **deny** | (Optional) Specifies that if the match criteria are met for the route map, the route is not redistributed. |
| *map_tag* | Text for the route map tag; the text can be up to 57 characters in length. |
| **permit** | (Optional) Specifies that if the match criteria is met for this route map, the route is redistributed as controlled by the set actions. |
| *seq_num* | (Optional) Route map sequence number; valid values are from 0 to 65535. Indicates the position that a new route map will have in the list of route maps already configured with the same name. |

## Defaults

The defaults are as follows:

- **permit.**
- If you do not specify a *seq_num*, a *seq_num* of 10 is assigned to the first route map.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| Preexisting | This command was preexisting. |

## Usage Guidelines

The **route-map** command lets you redistribute routes.

The **route-map** global configuration command `and the **match** and **set** configuration commands define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has **match** and **set** commands that are associated with it. The **match** commands specify the match criteria that are the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions, which are the redistribution actions to perform if the criteria enforced by the **match** commands are met. The **no route-map** command deletes the route map.

The **match route-map** configuration command has multiple formats. You can enter the **match** commands in any order, and all **match** commands must pass to cause the route to be redistributed according to the set actions given with the **set** commands. The **no** form of the **match** commands removes the specified match criteria.

Use route maps when you want detailed control over how routes are redistributed between routing processes. You specify the destination routing protocol with the **router ospf** global configuration command. You specify the source routing protocol with the **redistribute** router configuration command.

When you pass routes through a route map, a route map can have several parts. Any route that does not match at least one match clause relating to a **route-map** command is ignored; the route is not advertised for outbound route maps and is not accepted for inbound route maps. To modify only some data, you must configure a second route map section with an explicit match specified.

The *seq_number* argument is as follows:

1. If you do not define an entry with the supplied tag, an entry is created with the *seq_number* argument set to 10.
2. If you define only one entry with the supplied tag, that entry becomes the default entry for the following **route-map** command. The *seq_number* argument of this entry is unchanged.
3. If you define more than one entry with the supplied tag, an error message is printed to indicate that the *seq_number* argument is required.

If the **no route-map** *map-tag* command is specified (with no *seq-num* argument), the whole route map is deleted (all **route-map** entries with the same *map-tag* text).

If the match criteria are not met, and you specify the **permit** keyword, the next route map with the same *map_tag* is tested. If a route passes none of the match criteria for the set of route maps sharing the same name, it is not redistributed by that set.

## Examples

The following example shows how to configure a route map in OSPF routing:

```
hostname(config)# route-map maptag1 permit 8
hostname(config-route-map)# set metric 5
hostname(config-route-map)# match metric 5
hostname(config-route-map)# show running-config route-map
route-map maptag1 permit 8
    set metric 5
    match metric 5
hostname(config-route-map)# exit
hostname(config)#
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure route-map** | Removes the conditions for redistributing the routes from one routing protocol into another routing protocol. |
| **match interface** | Distributes distribute any routes that have their next hop out one of the interfaces specified, |
| **router ospf** | Starts and configures an ospf routing process. |
| **set metric** | Specifies the metric value in the destination routing protocol for a route map. |
| **show running-config route-map** | Displays the information about the route map configuration. |

# router-id

To use a fixed router ID, use the **router-id** command in router configuration mode. To reset OSPF to use the previous router ID behavior, use the **no** form of this command.

**router-id** *addr*

**no router-id** [*addr*]

## Syntax Description

| | |
|---|---|
| *addr* | Router ID in IP address format. |

## Defaults

If not specified, the highest-level IP address on the security appliance is used as the router ID.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Router configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| Preexisting | This command was preexisting. |
| 8.0(2) | The processing order for this command was changed. The command is now processed before the **network** commands in an OSPF configuration. |

## Usage Guidelines

By default, the security appliance uses the highest-level IP address on an interface that is covered by a **network** command in the OSPF configuration. If the highest-level IP address is a private address, then that address is sent in hello packets and database definitions. To use a specific router ID, use the **router-id** command to specify a global address for the router ID.

Router IDs must be unique within an OSPF routing domain. If two routers in the same OSPF domain are using the same router ID, routing may not work correctly.

You should enter the **router-id** command before entering **network** commands in an OSPF configuration. This prevents possible conflicts with the default router ID generated by the security appliance. If you do have a conflict, you will receive the message:

```
ERROR: router-id addr in use by ospf process pid
```

To enter the conflicting ID, remove the **network** command that contains the IP address causing the conflict, enter the **router-id** command, and then re-enter the **network** command.

## Examples

The following example sets the router ID to 192.168.1.1:

```
hostname(config-router)# router-id 192.168.1.1
hostname(config-router)#
```

## Related Commands

| Command | Description |
|---|---|
| **router ospf** | Enters router configuration mode. |
| **show ospf** | Displays general information about the OSPF routing processes. |

# router eigrp

To start an EIGRP routing process and configure parameters for that process, use the **router eigrp** command in global configuration mode. To disable EIGRP routing, use the **no** form of this command.

**router eigrp** *as-number*

**no router eigrp** *as-number*

## Syntax Description

| | |
|---|---|
| *as-number* | Autonomous system number that identifies the routes to the other EIGRP routers. It is also used to tag the routing information. Valid values are from 1 to 65535. |

## Defaults

EIGRP routing is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was introduced. |

## Usage Guidelines

The **router eigrp** command creates an EIGRP routing process or enters router configuration mode for an existing EIGRP routing process. You can only create a single EIGRP routing process on the security appliance.

Use the following router configuration mode commands to configure the EIGRP routing processes:

- **auto-summary**—Enable/disable automatic route summarization.
- **default-information**—Enable/disable the reception and sending of default route information.
- **default-metric**—Define the default metrics for routes redistributed into the EIGRP routing process.
- **distance eigrp**—Configure the administrative distance for internal and external EIGRP routes.
- **distribute-list**—Filter the networks received and sent in routing updates.
- **eigrp log-neighbor-changes**—Enable/disable the logging of neighbor state changes.
- **eigrp log-neighbor-warnings**—Enable/disable the logging of neighbor warning messages.
- **eigrp router-id**—Creates a fixed router ID.
- **eigrp stub**—Configures the security appliance for stub EIGRP routing.
- **neighbor**—Statically define an EIGRP neighbor.

- **network**—Configure the networks that participate in the EIGRP routing process.
- **passive-interface**—Configure an interface to act as a passive interface.
- **redistribute**—Redistribute routes from other routing processes into EIGRP.

Use the following interface configuration mode commands to configure interface-specific EIGRP parameters:

- **authentication key eigrp**—Define the authentication key used for EIGRP message authentication.
- **authentication mode eigrp**—Define the authentication algorithm used for EIGRP message authentication.
- **delay**—Configure the delay metric for an interface.
- **hello-interval eigrp**—Change the interval at which EIGRP hello packets are sent out of an interface.
- **hold-time eigrp**—Change the hold time advertised by the security appliance.
- **split-horizon eigrp**—Enable/disable EIGRP split-horizon on an interface.
- **summary-address eigrp**—Manually define a summary address.

## Examples

The following example shows how to enter the configuration mode for the EIGRP routing process with the autonomous system number 100:

```
hostname(config)# router eigrp 100
hostname(config-router)#
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure eigrp** | Clears the EIGRP router configuration mode commands from the running configuration. |
| **show running-config router eigrp** | Displays the EIGRP router configuration mode commands in the running configuration. |

# router ospf

To start an OSPF routing process and configure parameters for that process, use the **router ospf** command in global configuration mode. To disable OSPF routing, use the **no** form of this command.

**router ospf** *pid*

**no router ospf** *pid*

## Syntax Description

| | |
|---|---|
| *pid* | Internally used identification parameter for an OSPF routing process; valid values are from 1 to 65535. The *pid* does not need to match the ID of OSPF processes on other routers. |

## Defaults

OSPF routing is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| Preexisting | This command was preexisting. |

## Usage Guidelines

The **router ospf** command is the global configuration command for OSPF routing processes running on the security appliance. Once you enter the **router ospf** command, the command prompt appears as (config-router)#, indicating that you are in router configuration mode.

When using the **no router ospf** command, you do not need to specify optional arguments unless they provide necessary information. The **no router ospf** command terminates the OSPF routing process specified by its *pid.* You assign the *pid* locally on the security appliance. You must assign a unique value for each OSPF routing process.

The **router ospf** command is used with the following OSPF-specific commands to configure OSPF routing processes:

- **area**—Configures a regular OSPF area.
- **compatible rfc1583**—Restores the method used to calculate summary route costs per RFC 1583.
- **default-information originate**—Generates a default external route into an OSPF routing domain.
- **distance**—Defines the OSPF route administrative distances based on the route type.
- **ignore**—Suppresses the sending of syslog messages when the router receives a link-state advertisement (LSA) for type 6 Multicast OSPF (MOSPF) packets.

- **log-adj-changes**—Configures the router to send a syslog message when an OSPF neighbor goes up or down.
- **neighbor**—Specifies a neighbor router. Used to allow adjacency to be established over VPN tunnels.
- **network**—Defines the interfaces on which OSPF runs and the area ID for those interfaces.
- **redistribute**—Configures the redistribution of routes from one routing domain to another according to the parameters specified.
- **router-id**—Creates a fixed router ID.
- **summary-address**—Creates the aggregate addresses for OSPF.
- **timers lsa-group-pacing**—OSPF LSA group pacing timer (interval between group of LSA being refreshed or max-aged).
- **timers spf**—Delay between receiving a change to the SPF calculation.

## Examples

The following example shows how to enter the configuration mode for the OSPF routing process numbered 5:

```
hostname(config)# router ospf 5
hostname(config-router)#
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure router** | Clears the OSPF router commands from the running configuration. |
| **show running-config router ospf** | Displays the OSPF router commands in the running configuration. |

# router rip

To start a RIP routing process and configure parameters for that process, use the **router rip** command in global configuration mode. To disable the RIP routing process, use the **no** form of this command.

**router rip**

**no router rip**

## Syntax Description

This command has no arguments or keywords.

## Defaults

RIP routing is disabled.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| Command Mode | Routed | Transparent | Single | Context | System |
| Global configuration | • | — | • | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Usage Guidelines

The **router rip** command is the global configuration command for configuring the RIP routing processes on the security appliance. You can only configure one RIP process on the security appliance. The **no router rip** command terminates the RIP routing process and removes all router configuration for that process.

When you enter the **router rip** command the command prompt changes to `hostname(config-router)#`, indicating that you are in router configuration mode.

The **router rip** command is used with the following router configuration commands to configure RIP routing processes:

- **auto-summary**—Enable/disable automatic summarization of routes.
- **default-information originate**—Distribute a default route.
- **distribute-list in**—Filter networks in incoming routing updates.
- **distribute-list out**—Filter networks in outgoing routing updates.
- **network**—Add/remove interfaces from the routing process.
- **passive-interface**—Set specific interfaces to passive mode.
- **redistribute**—Redistribute routes from other routing processes into the RIP routing process.
- **version**—Set the RIP protocol version used by the security appliance.

Additionally, you can use the following commands in interface configuration mode to configure RIP properties on a per-interface basis:

- **rip authentication key**—Set an authentication key.
- **rip authentication mode**—Set the type of authentication used by RIP Version 2.
- **rip send version**—Set the version of RIP used to send updates out of the interface. This overrides the version set in global router configuration mode, if any.
- **rip receive version**—Set the version of RIP accepted by the interface. This overrides the version set in global router configuration mode, if any.

RIP is not supported under transparent mode. By default, the security appliance denies all RIP broadcast and multicast packets. To permit these RIP messages to pass through a security appliance operating in transparent mode you must define access list entries to permit this traffic. For example, to permit RIP version 2 traffic through the security appliance, create an access list entry such as `access-list myriplist extended permit ip any host 224.0.0.9`. To permit RIP version 1 broadcasts, create an access list entry such as `access-list myriplist extended permit udp any any eq rip`. Apply these access list entries to the appropriate interface using the **access-group** command.

You can enable both RIP and OSPF routing on the security appliance at the same time.

## Examples

The following example shows how to enter the configuration mode for the OSPF routing process numbered 5:

```
hostname(config)# router rip
hostname(config-router)# network 10.0.0.0
hostname(config-router)# version 2
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure router rip** | Clears the RIP router commands from the running configuration. |
| **show running-config router rip** | Displays the RIP router commands in the running configuration. |

# rtp-conformance

To check RTP packets flowing on the pinholes for protocol conformance in H.323 and SIP, use the **rtp-conformance** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

**rtp-conformance [enforce-payloadtype]**

**no rtp-conformance [enforce-payloadtype]**

## Syntax Description

| | |
|---|---|
| **enforce-payloadtype** | Enforces payload type to be audio/video based on the signaling exchange. |

## Defaults

No default behavior or values.

## Command Modes

The following table shows the modes in which you can enter the command:

| | Firewall Mode | | Security Context | | |
|---|---|---|---|---|---|
| | | | | Multiple | |
| **Command Mode** | **Routed** | **Transparent** | **Single** | **Context** | **System** |
| Parameters configuration | • | • | • | • | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was introduced. |

## Examples

The following example shows how to check RTP packets flowing on the pinholes for protocol conformance on an H.323 call:

```
hostname(config)# policy-map type inspect h323 h323_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# rtp-conformance
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Identifies a class map name in the policy map. |
| **class-map type inspect** | Creates an inspection class map to match traffic specific to an application. |
| **debug rtp** | Displays debug information and error messages for RTP packets associated with H.323 and SIP inspection. |
| **policy-map** | Creates a Layer 3/4 policy map. |
| **show running-config policy-map** | Display all current policy map configurations. |