



queue-limit through routerospf 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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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 queuing 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.

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
hostname(priority-queue)#
```

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 queued on a TCP stream, use the **queue-limit** command in tcp-map configuration mode. To remove this specification, use the **no** form of this command.

queue-limit *pkt_num*

no queue-limit *pkt_num*

Syntax Description

<i>pkt_num</i>	Specifies the maximum number of out-of-order packets that can be queued for a TCP connection before they are dropped. For ASA, the range is 0 to 250 and the default is 0. For PIX, the packet number is 3 and cannot be changed.
----------------	---

Defaults

The default maximum number of packets is 0 for the ASA platform. For PIX, the number is 3 and cannot be changed.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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 **queue-limit** command in tcp-map configuration mode to enable TCP packet ordering on any TCP connection or change the queue limit for connections that are ordered by default.

Packets will be ordered on TCP connections if any of the following features have been enabled: inspect, IDS feature, or TCP check-retransmission. The default packet queue limit for connections that are ordered is two per flow. For all other TCP connections, packets are forwarded as received, including out-of-order packets. To enable TCP packet ordering on any TCP connection or change the queue limit for connections that are ordered, use the **queue-limit** command. Enabling this feature results in out-of-order packets being queued until they can be forwarded or a fixed amount of time. Hence, memory usage is increased due to packet buffering.

Examples

The following example shows how to enable TCP packet ordering on all telnet connections:

```
hostname(config)# tcp-map tmap
hostname(config-tcp-map)# queue-limit 8
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	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.

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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple 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

<i>string</i>	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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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 common 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-with-expiry

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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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.

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
hostname(config-tunnel-ipsec)#
```

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.

reactivation-mode

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

reactivation-mode depletion [*deadtime minutes*]

reactivation-mode timed

no reactivation-mode

Syntax Description

deadtime <i>minutes</i>	(Optional) Specifies the amount of time that elapses between the disabling of the last server in the group and the subsequent re-enabling of all servers.
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. The supported range of values for deadtime is 0-1440 minutes.

Command Modes

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

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

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 a TACACS+ AAA server named “svrgrp1” to use the depletion reactivation mode, with a deadtime of 15 minutes:

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

The following example configures a TACACS+ AAA server named “svrgrp1” 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

redistribute

To redistribute routes from one routing domain into another routing domain, 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 } [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] }} | 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.
external <i>type</i>	Specifies the OSPF metric routes that are external to a specified autonomous system; valid values are 1 or 2 .
internal <i>type</i>	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 <i>metric_value</i>	(Optional) Specifies the OSPF default metric value from 0 to 16777214.
metric-type <i>metric_type</i>	(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 <i>type</i>	Specifies the OSPF metric type for routes that are external to a not-so-stubby area (NSSA); valid values are 1 or 2 .
ospf <i>pid</i>	Used to redistribute an OSPF routing process into the current OSPF routing process. The <i>pid</i> specifies the internally used identification parameter for an OSPF routing process; valid values are from 1 to 65535.
route-map <i>map_name</i>	(Optional) Name of the route map to apply.
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 <i>tag_value</i>	(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

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Examples

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

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

Related Commands


Command	Description
router ospf	Enters router configuration mode.
show running-config router	Displays the commands in the global router configuration.

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 <i>hh:mm</i>	(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.
<i>day</i>	(Optional) Number of the day in the range from 1 to 31.
in [<i>hh:mm</i>]	(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 [<i>hh:mm</i>]	(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.
<i>month</i>	(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 <i>text</i>	(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.
	 <p>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.</p>
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:

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

Command History

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

Usage Guidelines

The `reload` 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
...
```

■ reload

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:

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

Command History

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

Usage Guidelines

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] [flash:] *source-path* [flash:] *destination-path*

Syntax Description	
/noconfirm	(Optional) Suppresses the confirmation prompt.
<i>destination-path</i>	Specifies the path of the destination file.
flash:	(Optional) Specifies the internal Flash memory, followed by a colon.
<i>source-path</i>	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:

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

Command History	Release	Modification
	7.0(1)	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.

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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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 are 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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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
hostname(config-ftp-map)#
```

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-method

To restrict HTTP traffic based on the HTTP request method, use the **request-method** command in HTTP map configuration mode, which is accessible using the **http-map** command. To disable this feature, use the **no** form of the command.

```
request-method { { ext ext_methods | default } | { rfc rfc_methods | default } } action { allow | reset | drop } [log]
```

```
no request-method { ext ext_methods | rfc rfc_methods } action { allow | reset | drop } [log]
```

Syntax Description

action	Identifies the action taken when a message fails this command inspection.
allow	Allows the message.
default	Specifies the default action taken by the security appliance when the traffic contains a supported request method that is not on a configured list.
drop	Closes the connection.
ext	Specifies extension methods.
<i>ext_methods</i>	Identifies one of the extended methods you want to allow to pass through the security appliance.
log	(Optional) Generates a syslog.
reset	Sends a TCP reset message to client and server.
rfc	Specifies RFC 2616 supported methods.
<i>rfc_methods</i>	Identifies one of the RFC methods you want to allow to pass through the security appliance (see Table 23-1).

Defaults

This command is disabled by default. When the command is enabled and a supported request method is not specified, the default action is to allow the connection without logging. To change the default action, use the **default** keyword and specify a different default action.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
HTTP map configuration	•	•	•	•	—

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

When you enable the **request-method** command, the security appliance applies the specified action to HTTP connections for each supported and configured request method.

The security appliance applies the **default** action to all traffic that does *not* match the request methods on the configured list. The **default** action is to **allow** connections without logging. Given this preconfigured default action, if you specify one or more request methods with the action of **drop** and **log**, the security appliance drops connections containing the configured request methods, logs each connection, and allows all connections containing other supported request methods.

If you want to configure a more restrictive policy, change the default action to **drop** (or **reset**) and **log** (if you want to log the event). Then configure each permitted method with the **allow** action.

Enter the **request-method** command once for each setting you wish to apply. You use one instance of the **request-method** command to change the default action or to add a single request method to the list of configured methods.

When you use the **no** form of the command to remove a request method from the list of configured methods, any characters in the command line after the request method keyword are ignored.

[Table 23-1](#) lists the methods defined in RFC 2616 that you can add to the list of configured methods:

Table 23-1 RFC 2616 Methods

Method	Description
connect	Used with a proxy that can dynamically switch to being a tunnel (for example SSL tunneling).
delete	Requests that the origin server delete the resource identified by the Request-URI.
get	Retrieves whatever information or object is identified by the Request-URI.
head	Identical to GET except that the server does not return a message-body in the response.
options	Represents a request for information about the communication options available on server identified by the Request-URI.
post	Request that the origin server accept the object enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.
put	Requests that the enclosed object be stored under the supplied Request-URI.
trace	Invokes a remote, application-layer loop-back of the request message.

Examples

The following example provides a permissive policy, using the preconfigured default, which allows all supported request methods that are not specifically prohibited.

```
hostname(config)# http-map inbound_http
hostname(config-http-map)# request-method rfc options drop log
hostname(config-http-map)# request-method rfc post drop log
hostname(config-http-map)
```

In this example, only the **options** and **post** request methods are dropped and the events are logged.

The following example provides a restrictive policy, with the default action changed to **reset** the connection and **log** the event for any request method that is not specifically allowed.

```
hostname(config)# http-map inbound_http
hostname(config-http-map)# request-method rfc default action reset log
hostname(config-http-map)# request-method rfc get allow
```

```
hostname(config-http-map) # request-method rfc put allow
hostname(config-http-map) #
```

In this case, the **get** and **put** request methods are allowed. When traffic is detected that uses any other methods, the security appliance resets the connection and creates a syslog entry.

Related Commands

Commands	Description
class-map	Defines the traffic class to which to apply security actions.
debug appfw	Displays detailed information about traffic associated with enhanced HTTP inspection.
http-map	Defines an HTTP map for configuring enhanced HTTP inspection.
inspect http	Applies a specific HTTP map to use for application inspection.
policy-map	Associates a class map with specific security actions.

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

<i>max_requests</i>	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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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 gtp-policy
hostname(config-gtpmap)# request-queue-size 300
hostname(config-gtpmap)#
```

Related Commands

Commands	Description
clear service-policy inspect gtp	Clears global GTP statistics.
debug gtp	Displays detailed information about GTP inspection.
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-sso-siteminder configuration mode. This is an SSO with CA SiteMinder command.

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

request-timeout *seconds*

no request-timeout

Syntax Description

<i>seconds</i>	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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Webvpn-sso-siteminder 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 reentering a username and password more than once. The security appliance currently supports the Computer Associates eTrust SiteMinder SSO server (formerly Netegrity SiteMinder).

Once you have configured the security appliance to support SSO authentication, you can then optionally 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-sso-siteminder configuration mode, configures an authentication timeout at ten seconds for the SiteMinder SSO server “example”:

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

request-timeout

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

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 an SSO server.
show webvpn sso-server	Displays the operating statistics for an SSO server.
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 SSO authentication requests.

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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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
hostname(config)#

```

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.

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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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
hostname(config-dns-server-group)#
```

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.

retries

Command	Description

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

<i>seconds</i>	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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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.

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.
<i>integer</i>	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.
<i>resource name</i>	(Optional) Specifies the application or resource to which the rule applies. Maximum 128 bytes.
<i>string</i>	Specifies the name of the application or resource to match that can contain a regular expression. You can use the following wildcards: Specifies a pattern to match that can contain a regular expression. You can use the following wildcards: <ul style="list-style-type: none"> * — Matches everything. You cannot use this wildcard by itself. It must accompany an alphanumeric string. ? —Matches any single character. [!seq] — Matches any character not in sequence. [seq] — Matches any character in sequence. 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		
	Routed	Transparent	Single	Multiple	
			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-webvpn)# rewrite order 2 disable resource-mask *cisco.com/*
hostname(config-webvpn)#
```

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 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 | disable}

no re-xauth

Syntax Description

disable	Disables reauthentication on IKE rekey
enable	Enables reauthentication on IKE rekey

Defaults

Reauthentication on IKE rekey is disabled.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

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.

If the configured rekey interval is very short, users might find the repeated authorization requests inconvenient. In this case, disable reauthentication. 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
```

rmdir

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

```
rmdir [/noconfirm] [flash:]path
```

Syntax Description

noconfirm	(Optional) Suppresses the confirmation prompt.
flash:	(Optional) Specifies the nonremovable internal Flash, followed by a colon.
<i>path</i>	(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:

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

Command History

Release	Modification
7.0(1)	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* | **tunneled**]

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

Syntax Description

<i>gateway_ip</i>	Specifies the IP address of the gateway router (the next-hop address for this route).
	Note The <i>gateway_ip</i> argument is optional in transparent mode.
<i>interface_name</i>	Internal or external network interface name.
<i>ip_address</i>	Internal or external network IP address.
<i>metric</i>	(Optional) The administrative distance for this route. Valid values range from 1 to 255. The default value is 1.
<i>netmask</i>	Specifies a network mask to apply to <i>ip_address</i> .
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:

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

Command History

Release	Modification
Preexisting	This command was preexisting.

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 encrypted traffic that arrives on the security appliance and cannot be routed using learned or static routes is sent to this route. Otherwise, if the traffic is not encrypted, the standard default route entry is used. 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
```

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.
<i>map_tag</i>	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.
<i>seq_num</i>	(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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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:

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

If the highest-level IP address on the security appliance is a private address, then this address is sent in hello packets and database definitions. To prevent this situation, use the **router-id** command to specify a global address for the router ID.

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 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:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				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.

You cannot configure OSPF when RIP is configured on the security appliance.

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.