

mac-address-table aging-time through multicast-routing Commands

mac-address-table aging-time

To set the timeout for MAC address table entries, use the **mac-address-table aging-time** command in global configuration mode. To restore the default value of 5 minutes, use the **no** form of this command.

mac-address-table aging-time *timeout_value*

no mac-address-table aging-time

Syntax Description	<i>timeout_value</i>	The time a MAC address entry stays in the MAC address table before timing out, between 5 and 720 minutes (12 hours). 5 minutes is the default.
---------------------------	----------------------	--

Defaults The default timeout is 5 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
Global configuration	—	•	•	•	—

Command History	Release	Modification
	2.2(1)	This command was introduced.

Examples The following example sets the MAC address timeout to 10 minutes:

```
hostname(config)# mac-address-timeout aging time 10
```

Related Commands	Command	Description
	arp-inspection	Enables ARP inspection, which compares ARP packets to static ARP entries.
	firewall transparent	Sets the firewall mode to transparent.
	mac-address-table static	Adds static MAC address entries to the MAC address table.
	mac-learn	Disables MAC address learning.
	show mac-address-table	Shows the MAC address table, including dynamic and static entries.

mac-address-table static

To add a static entry to the MAC address table, use the **mac-address-table static** command in global configuration mode. To remove a static entry, use the **no** form of this command.

```
mac-address-table static interface_name mac_address
```

```
no mac-address-table static interface_name mac_address
```

Syntax Description

<i>interface_name</i>	Sets the source interface.
<i>mac_address</i>	Sets the MAC address you want to add to the table.

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
2.2(1)	This command was introduced.

Usage Guidelines

Normally, MAC addresses are added to the MAC address table dynamically as traffic from a particular MAC address enters an interface. You can add static MAC addresses to the MAC address table if desired. One benefit to adding static entries is to guard against MAC spoofing. If a client with the same MAC address as a static entry attempts to send traffic to an interface that does not match the static entry, then the FWSM drops the traffic and generates a system message.

Examples

The following example adds a static MAC address entry to the MAC address table:

```
hostname(config)# mac-address-table static inside 0010.7cbe.6101
```

Related Commands

Command	Description
arp	Adds a static ARP entry.
firewall transparent	Sets the firewall mode to transparent.
mac-address-table aging-time	Sets the timeout for dynamic MAC address entries.

Command	Description
mac-learn	Disables MAC address learning.
show mac-address-table	Shows MAC address table entries.

mac-learn

To disable MAC address learning for an interface, use the **mac-learn** command in global configuration mode. To reenable MAC address learning, use the **no** form of this command.

mac-learn *interface_name* **disable**

no mac-learn *interface_name* **disable**

Syntax Description

<i>interface_name</i>	Sets the interface on which you want to disable MAC learning.
disable	Disables MAC learning.

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
2.2(1)	This command was introduced.

Usage Guidelines

By default, each interface automatically learns the MAC addresses of entering traffic, and the FWSM adds corresponding entries to the MAC address table. You can disable MAC address learning if desired.

Examples

The following example disables MAC learning on the outside interface:

```
hostname(config)# mac-learn outside disable
```

Related Commands

Command	Description
clear configure mac-learn	Sets the mac-learn configuration to the default.
firewall transparent	Sets the firewall mode to transparent.
mac-address-table static	Adds static MAC address entries to the MAC address table.

Command	Description
show mac-address-table	Shows the MAC address table, including dynamic and static entries.
show running-config mac-learn	Shows the mac-learn configuration.

mac-list

To specify a list of MAC addresses to be used for MAC-based authentication, use the **mac-list** command in global configuration mode. To disable the use of a list of MAC addresses, use the **no** form of this command. The **mac-list** command adds a list of MAC addresses using a first-match search.

```
mac-list id {deny | permit} mac macmask
```

```
no mac-list id {deny | permit} mac macmask
```

Syntax Description

deny	Indicates that traffic matching these criteria is <i>not</i> included in the MAC list and is subject to both authentication and authorization.
<i>id</i>	Specifies an alphanumeric name to the MAC access list.
<i>mac</i>	Specifies the source MAC address in 12-digit hexadecimal form; that is, nnnn.nnnn.nnnn
<i>macmask</i>	Specifies and applies the netmask to <i>mac</i> and allows the grouping of MAC addresses.
permit	Indicates that traffic matching these criteria is included in the MAC list and is exempt from both authentication and authorization.

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
Global configuration	•	•	—	—	•

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To group a set of MAC addresses, enter the **mac-list** command as many times as needed with the same *id* value. Configure the MAC access list number using the **mac-list** command before using the **aaa mac-exempt** command.

Only AAA exemption is provided. Authorization is automatically exempted for MAC addresses for which authentication is exempted. Other types of AAA with **mac-list** are not supported.

Examples

The following example shows how to configure a MAC address list:

```
hostname(config)# mac-list adc permit 00a0.ca5d.0282 ffff.ffff.ffff
```

```

hostname(config)# mac-list adc deny 00a1.ca5d.0282 ffff.ffff.ffff
hostname(config)# mac-list ac permit 0050.54ff.0000 ffff.ffff.0000
hostname(config)# mac-list ac deny 0061.54ff.b440 ffff.ffff.ffff
hostname(config)# mac-list ac deny 0072.54ff.b440 ffff.ffff.ffff

```

Related Commands

Command	Description
aaa authentication	Enable, disable, or view LOCAL, TACACS+, or RADIUS user authentication, on a server designated by the aaa-server command, or ASDM user authentication.
aaa authorization	Enable or disable LOCAL or TACACS+ user authorization services.
aaa mac-exempt	Exempt a list of MAC addresses from authentication and authorization.
clear configure mac-list	Remove a list of MAC addresses previously specified the mac-list command with the indicated MAC list number.
show running-config mac-list	Display a list of MAC addresses previously specified in the mac-list command with the indicated MAC list number.

management-access

To allow management access to an interface other than the one you entered the FWSM from, use the **management-access** command in global configuration mode. To disable this access, use the **no** form of this command.

management-access *mgmt_if*

no management-access *mgmt_if*

Syntax Description

mgmt_if Specifies the name of the management interface you want to access when entering the FWSM from another interface.

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
3.1(1)	This command was introduced.

Usage Guidelines

This command allows you to connect to an interface other than the one you entered the FWSM from. For example, if you enter the FWSM from the outside interface, this command lets you connect to the inside interface using Telnet; or you can ping the inside interface when entering from the outside interface.

You can define only one management interface.

The **management-access** command is supported for the following through an IPSec VPN tunnel only:

- SNMP polls to the management interface
- HTTPS requests to the management interface
- ASDM access to the management interface
- Telnet access to the management interface
- SSH access to the management interface
- Ping to the management interface
- Syslog polls to the management interface
- NTP requests the management interface

Examples

The following example shows how to configure a firewall interface named “inside” as the management access interface:

```
hostname(config)# management-access inside
hostname(config)# show management-access
management-access inside
```

Related Commands

Command	Description
clear configure management-access	Removes the configuration of an interface for management access of the FWSM.
show management-access	Displays the name of the interface configured for management access.

mask-syst-reply

To hide the FTP server response from clients, use the **mask-syst-reply** 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.

mask-syst-reply

no mask-syst-reply

Syntax Description

This command has no arguments or keywords.

Defaults

This command is enabled 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
FTP map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Use the **mask-syst-reply** command with strict FTP inspection to protect the FTP server system from clients. After enabling this command, the servers replies to the **syst** command are replaced by a series of Xs.

Examples

The following example causes the FWSM to replace the FTP server replies to the **syst** command with Xs:

```
hostname(config)# ftp-map inbound_ftp
hostname(config-ftp-map)# mask-syst-reply
hostname(config-ftp-map)# exit
```

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.
policy-map	Associates a class map with specific security actions.
request-command deny	Specifies FTP commands to disallow.

match access-list

To identify traffic using an access list in a class map, use the **match access-list** command in class-map configuration mode. To remove the access list, use the **no** form of this command.

match access-list {*acl-id...*}

no match access-list {*acl-id...*}

Syntax Description

acl-id Specifies the name of an ACL to be used as match criteria. When a packet does not match an entry in the ACL, the match result is a no-match. When a packet matches an entry in an ACL, and if it is a permit entry, the match result is a match. Otherwise, if it matches a deny ACL entry, the match result is no-match.

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

You can specify one or more access lists to identify specific types of traffic using the **match access-list** command. The **permit** statement in an access control entry causes the traffic to be included, while a **deny** statement causes the traffic to be excluded from the traffic class map.

Examples

The following example shows how to define a traffic class using a class map and the **match access-list** command:

```
hostname(config)# access-list ftp_acl extended permit tcp any any eq 21  
hostname(config)# class-map ftp_port  
hostname(config-cmap)# match access-list ftp_acl
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes of the traffic map definitions.
match any	Includes all traffic in the class map.
match port	Identifies a specific port number in a class map.
show running-config class-map	Displays the information about the class map configuration.

match any

To include all traffic in a class map, use the **match any** command in class-map configuration mode. To remove this specification, use the **no match any** form of this command.

match any

no match any

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

All packets will be matched using the **match any** command (as in the default class map, class-default).

Examples

This example shows how to define a traffic class using a class map and the **match any** command:

```
hostname(config)# class-map cmap
hostname(config-cmap)# match any
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes all of the traffic map definitions.
match access-list	Identifies access list traffic in a class map.
match rtp	Identifies a specific RTP port in a class map.
show running-config class-map	Displays the information about the class map configuration.

match default-inspection-traffic

To specify default traffic for the inspect commands in a class map, use the **match default-inspection-traffic** command in class-map configuration mode. To remove this specification, use the **no** form of this command.

match default-inspection-traffic

no match default-inspection-traffic

Syntax Description

This command has no arguments or keywords.

Defaults

See the “Usage Guidelines” section for the default traffic of each inspection.

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

Using the **match default-inspection-traffic** command, you can match default traffic for the individual **inspect** commands. The **match default-inspection-traffic** command can be used in conjunction with one other match command, which is typically an access-list in the form of **permit ip src-ip dst-ip**.

The rule for combining a second **match** command with the **match default-inspection-traffic** command is to specify the protocol and port information using the **match default-inspection-traffic** command and specify all other information (such as IP addresses) using the second **match** command. Any protocol or port information specified in the second **match** command is ignored with respect to the **inspect** commands.

For instance, port 65535 specified in the example below is ignored:

```
hostname(config)# class-map cmap
hostname(config-cmap)# match default-inspection-traffic
hostname(config-cmap)# match port 65535
```

Default traffic for inspections are as follows:

Inspection Type	Protocol Type	Source Port	Destination Port
ctiqbe	tcp	N/A	1748
dns	udp	53	53
ftp	tcp	N/A	21
gtp	udp	2123,3386	2123,3386
h323 h225	tcp	N/A	1720
h323 ras	udp	N/A	1718-1719
http	tcp	N/A	80
icmp	icmp	N/A	N/A
ils	tcp	N/A	389
mgcp	udp	2427,2727	2427,2727
netbios	udp	137-138	N/A
rpc	udp	111	111
rsh	tcp	N/A	514
rtsp	tcp	N/A	554
sip	tcp,udp	N/A	5060
skinny	tcp	N/A	2000
smtp	tcp	N/A	25
sqlnet	tcp	N/A	1521
tftp	udp	N/A	69
xdmcp	udp	177	177

Examples

The following example shows how to define a traffic class using a class map and the **match default-inspection-traffic** command:

```
hostname(config)# class-map cmap
hostname(config-cmap)# match default-inspection-traffic
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes all of the traffic map definitions.
match access-list	Identifies access list traffic within a class map.
match any	Includes all traffic in the class map.
show running-config class-map	Displays the information about the class map configuration.

match dscp

To identify the IETF-defined DSCP value (in an IP header) in a class map, use the **match dscp** command in class-map configuration mode. To remove this specification, use the **no** form of this command.

```
match dscp {values}
```

```
no match dscp {values}
```

Syntax Description

<i>values</i>	Specifies up to eight different the IETF-defined DSCP values in the IP header. Range is 0 to 63.
---------------	--

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

Using the **match dscp** command, you can match the IETF-defined DSCP values in the IP header.

Examples

The following example shows how to define a traffic class using a class map and the **match dscp** command:

```
hostname(config)# class-map cmap
hostname(config-cmap)# match dscp af43 cs1 ef
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes all of the traffic map definitions.
match access-list	Identifies access list traffic within a class map.
match port	Specifies the TCP/UDP ports as the comparison criteria for packets received on that interface.
show running-config class-map	Displays the information about the class map configuration.

match interface

To distribute any routes that have their next hop out one of the interfaces specified, use the **match interface** command in route-map configuration mode. To remove the match interface entry, use the **no** form of this command.

match interface *interface-name...*

no match interface *interface-name...*

Syntax Description

interface-name Name of the interface as specified by the **nameif** command. You can specify multiple interface names.

Defaults

No match interfaces are defined.

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
Route-map configuration	•	—	•	—	—

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the interface-type interface-number arguments.

The **route-map global** configuration command and the **match** and **set** configuration commands let you 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—the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions—the particular redistribution actions to perform if the criteria that is 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 give the **match** commands in any order. All **match** commands must “pass” to cause the route to be redistributed according to the set actions that are given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria. If there is more than one interface specified in the **match** command, then the **no match interface interface-name** can be used to remove a single interface.

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. If you want to modify only some data, you must configure a second route map section and specify an explicit match.

Examples

The following example shows that the routes with their next hop outside is distributed:

```
hostname(config)# route-map name
hostname(config-route-map)# match interface outside
```

Related Commands

Command	Description
match ip next-hop	Distributes any routes that have a next-hop router address that is passed by one of the access lists specified.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address that is specified by the access lists.
match metric	Redistributes routes with the metric specified.
route-map	Defines the conditions for redistributing routes from one routing protocol into another.
set metric	Specifies the metric value in the destination routing protocol for a route map.

match ip address

To redistribute any routes that have a route address or match packet that is passed by one of the access lists specified, use the **match ip address** command in route-map configuration mode. To restore the default settings, use the **no** form of this command.

```
match ip address {acl...}
```

```
no match ip address {acl...}
```

Syntax Description

acl Specifies an ACL by name. You can specify multiple ACLs.

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
Route-map configuration	•	—	•	—	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **route-map global** configuration command and the **match** and **set** configuration commands let you 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—the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions—the particular redistribution actions to perform if the criteria that is enforced by the **match** commands are met. The **no route-map** command deletes the route map.

Examples

The following example shows how to redistribute internal routes:

```
hostname(config)# route-map name
hostname(config-route-map)# match ip address acl_dmz1 acl_dmz2
```

Related Commands

Command	Description
match interface	Distributes distribute any routes that have their next hop out one of the interfaces specified.
match ip next-hop	Distributes any routes that have a next-hop router address that is passed by one of the access lists specified.
match metric	Redistributes routes with the metric specified.
route-map	Defines the conditions for redistributing routes from one routing protocol into another.
set metric	Specifies the metric value in the destination routing protocol for a route map.

match ip next-hop

To redistribute any routes that have a next-hop router address that is passed by one of the access lists specified, use the **match ip next-hop** command in route-map configuration mode. To remove the next-hop entry, use the **no** form of this command.

```
match ip next-hop {acl... | prefix-list prefix_list}
```

```
no match ip next-hop {acl... | prefix-list prefix_list}
```

Syntax Description

<i>acl</i>	Name of an ACL. You can specify multiple ACLs.
prefix-list prefix_list	Name of prefix list.

Defaults

Routes are distributed freely, without being required to match a next-hop address.

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
Route-map configuration	•	—	•	—	—

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the access-list-name argument.

The **route-map global** configuration command and the **match** and **set** configuration commands let you 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—the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions—the particular redistribution actions to perform if the criteria that is 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. All **match** commands must “pass” to cause the route to be redistributed according to the set actions given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

When you are passing 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. To modify only some data, you must configure a second route map section and specify an explicit match.

Examples

The following example shows how to distribute routes that have a next-hop router address passed by access list `acl_dmz1` or `acl_dmz2`:

```
hostname# route-map name
hostname(config-route-map)# match ip next-hop acl_dmz1 acl_dmz2
```

Related Commands

Command	Description
match interface	Distributes distribute any routes that have their next hop out one of the interfaces specified.
match ip next-hop	Distributes any routes that have a next-hop router address that is passed by one of the access lists specified.
match metric	Redistributes routes with the metric specified.
route-map	Defines the conditions for redistributing routes from one routing protocol into another.
set metric	Specifies the metric value in the destination routing protocol for a route map.

match ip route-source

To redistribute routes that have been advertised by routers and access servers at the address that is specified by the access lists, use the **match ip route-source** command in the route-map configuration mode. To remove the next-hop entry, use the **no** form of this command.

```
match ip route-source {acl... | prefix-list prefix_list}
```

```
no match ip route-source {acl... | prefix-list prefix_list}
```

Syntax Description

<i>acl</i>	Name of an ACL. You can specify multiple ACLs.
<i>prefix_list</i>	Name of prefix list.

Defaults

No filtering on a route source.

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
Route-map configuration	•	—	•	—	—

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the access-list-name argument.

The **route-map global** configuration command and the **match** and **set** configuration commands let you 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—the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions—the particular redistribution actions to perform if the criteria that is 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. All **match** commands must “pass” to cause the route to be redistributed according to the set actions given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

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. To modify only some data, you must configure a second route map section and specify an explicit match. The next-hop and source-router address of the route are not the same in some situations.

Examples

The following example shows how to distribute routes that have been advertised by routers and access servers at the addresses specified by access lists `acl_dmz1` and `acl_dmz2`:

```
hostname(config)# route-map name
hostname(config-route-map)# match ip route-source acl_dmz1 acl_dmz2
```

Related Commands

Command	Description
match interface	Distributes distribute any routes that have their next hop out one of the interfaces specified.
match ip next-hop	Distributes any routes that have a next-hop router address that is passed by one of the access lists specified.
match metric	Redistributes routes with the metric specified.
route-map	Defines the conditions for redistributing routes from one routing protocol into another.
set metric	Specifies the metric value in the destination routing protocol for a route map.

match metric

To redistribute routes with the metric specified, use the **match metric** command in route-map configuration mode. To remove the entry, use the **no** form of this command.

match metric *number*

no match metric *number*

Syntax Description

number Route metric; valid values are from 0 to 4294967295.

Defaults

No filtering on a metric value.

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
Route-map configuration	•	—	•	—	—

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

The **route-map global** configuration command and the **match** and **set** configuration commands let you 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—the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions—the particular redistribution actions to perform if the criteria that is 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. The **match** commands can be given 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** forms of the **match** commands remove the specified match criteria.

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. To modify only some data, you must configure a second route map section and specify an explicit match.

Examples

The following example shows how to redistribute routes with the metric 5:

```
hostname(config)# route-map name
hostname(config-route-map)# match metric 5
```

Related Commands	Command	Description
	match interface	Distributes distribute any routes that have their next hop out one of the interfaces specified.
	match ip next-hop	Distributes any routes that have a next-hop router address that is passed by one of the access lists specified.
	route-map	Defines the conditions for redistributing routes from one routing protocol into another.
	set metric	Specifies the metric value in the destination routing protocol for a route map.

match port

To identify a specific port number in a class map, use the **match port** command in class-map configuration mode. To remove this specification, use the **no** form of this command.

```
match port {tcp | udp} {eq eq_id | range beg_id end_id}
```

```
no match port {tcp | udp} {eq eq_id | range beg_id end_id}
```

Syntax Description

eq <i>eq_id</i>	Specifies a port name.
range <i>beg_id</i> <i>end_id</i>	Specifies beginning and ending port range values (1-65535).
tcp	Specifies a TCP port.
udp	Specifies a UDP port.

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

Use the **match port** command to specify a range of ports.

Examples

The following example shows how to define a traffic class using a class map and the **match port** command:

```
hostname(config)# class-map cmap
```

```
hostname(config-cmap) # match port tcp eq 8080
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes all of the traffic map definitions.
match access-list	Identifies access list traffic within a class map.
match any	Includes all traffic in the class map.
show running-config class-map	Displays the information about the class map configuration.

match precedence

To specify a precedence value in a class map, use the **match precedence** command in class-map configuration mode. To remove this specification, use the **no** form of this command.

match precedence *value*

no match precedence *value*

Syntax Description

value Specifies up to four precedence values separated by a space. Range is 0 to 7.

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

Use the **match precedence** command to specify the value represented by the TOS byte in the IP header.

Examples

The following example shows how to define a traffic class using a class map and the **match precedence** command:

```
hostname(config)# class-map cmap
hostname(config-cmap)# match precedence 1
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes all of the traffic map definitions.
match access-list	Identifies access list traffic within a class map.
match any	Includes all traffic in the class map.
show running-config class-map	Displays the information about the class map configuration.

match route-type

To redistribute routes of the specified type, use the **match route-type** command in route-map configuration mode. To remove the route type entry, use the **no** form of this command.

```
match route-type {local | internal | {external [type-1 | type-2]} | {nssa-external [type-1 | type-2]}}
```

```
no match route-type {local | internal | {external [type-1 | type-2]} | {nssa-external [type-1 | type-2]}}
```

Syntax Description

external	Match OSPF external routes (type 1 or type 2).
internal	Match OSPF intra-area and interarea routes.
local	Match a locally generated route.
nssa-external	Match OSPF NSSA external route (type 1 or type 2).
type-1	(Optional) Match only type 1 routes.
type-2	(Optional) Match only type 2 routes.

Defaults

This command is disabled 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
Route-map configuration	•	—	•	—	—

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

The **route-map global** configuration command and the **match** and **set** configuration commands let you 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—the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions—the particular redistribution actions to perform if the criteria that is 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. All **match** commands must “pass” to cause the route to be redistributed according to the set actions given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

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. To modify only some data, you must configure a second route map section and specify an explicit match.

Examples

The following example shows how to redistribute internal routes:

```
hostname(config)# route-map name
hostname(config-route-map)# match route-type internal
```

Related Commands

Command	Description
match interface	Distributes distribute any routes that have their next hop out one of the interfaces specified.
match ip next-hop	Distributes any routes that have a next-hop router address that is passed by one of the access lists specified.
match metric	Redistributes routes with the metric specified.
route-map	Defines the conditions for redistributing routes from one routing protocol into another.
set metric	Specifies the metric value in the destination routing protocol for a route map.

match rtp

To specify a UDP port range of even-number ports in a class map, use the **match rtp** command in class-map configuration mode. To remove this specification, use the **no** form of this command.

match rtp *starting_port range*

no match rtp *starting_port range*

Syntax Description

<i>starting_port</i>	Specifies lower bound of even-number UDP destination port. Range is 2000-65535
<i>range</i>	Specifies range of RTP ports. Range is 0-16383.

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
Class-map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **match** commands are used to identify the traffic included in the traffic class for a class map. They include different criteria to define the traffic included in a class-map. Define a traffic class using the **class-map** global configuration command as part of configuring a security feature using Modular Policy Framework. From class-map configuration mode, you can define the traffic to include in the class using the **match** command.

After a traffic class is applied to an interface, packets received on that interface are compared to the criteria defined by the **match** statements in the class map. If the packet matches the specified criteria, it is included in the traffic class and is subjected to any actions associated with that traffic class. Packets that do not match any of the criteria in any traffic class are assigned to the default traffic class.

Use the **match rtp** command to match RTP ports (even UDP port numbers between the *starting_port* and the *starting_port* plus the *range*).

Examples

The following example shows how to define a traffic class using a class map and the **match rtp** command:

```
hostname(config)# class-map cmap
```

```
hostname(config-cmap) # match rtp 20000 100
```

Related Commands

Command	Description
class-map	Applies a traffic class to an interface.
clear configure class-map	Removes all of the traffic map definitions.
match access-list	Identifies access list traffic within a class map.
match any	Includes all traffic in the class map.
show running-config class-map	Displays the information about the class map configuration.

max-failed-attempts

To specify the number of failed attempts allowed for any given server in the server group before that server is deactivated, use the **max-failed-attempts** command in AAA-server group mode. To remove this specification and revert to the default value, use the **no** form of this command:

```
max-failed-attempts number
```

```
no max-failed-attempts
```

Syntax Description

number An integer in the range 1-5, specifying the number of failed connection attempts allowed for any given server in the server group specified in a prior **aaa-server** command.

Defaults

The default value of *number* is 3.

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	This command was introduced.

Usage Guidelines

You must have configured the AAA server/group before issuing this command.

Examples

```
hostname(config)# aaa-server svrgrp1 protocol tacacs+
hostname(config-aaa-server-group)# max-failed-attempts 4
```

Related Commands

Command	Description
aaa-server <i>server-tag</i> protocol <i>protocol</i>	Enters AAA server group configuration mode so that you can configure AAA server parameters that are group-specific and common to all hosts in the group.

clear configure aaa-server	Removes all AAA server configuration.
show running-config aaa	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

max-header-length

To restrict HTTP traffic based on the HTTP header length, use the **max-header-length** command in HTTP map configuration mode, which is accessible using the **http-map** command. To remove this command, use the **no** form of this command.

```
max-header-length { request bytes [response bytes] | response bytes } action { allow | reset | drop }
[log]
```

```
no max-header-length { request bytes [response bytes] | response bytes } action { allow | reset |
drop } [log]
```

Syntax Description

action	The action taken when a message fails this command inspection.
allow	Allow the message.
drop	Closes the connection.
bytes	Number of bytes, range is 1 to 65535.
log	(Optional) Generate a syslog.
request	Request message.
reset	Send a TCP reset message to client and server.
response	(Optional) Response message.

Defaults

This command is disabled 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
HTTP map configuration	•	•	•	•	—

Command History

Release	Modification
3.1	This command was introduced.

Usage Guidelines

After enabling the **max-header-length** command, the FWSM only allows messages having an HTTP header within the configured limit and otherwise takes the specified action. Use the **action** keyword to cause the FWSM to reset the TCP connection and optionally create a syslog entry.

Examples

The following example restricts HTTP requests to those with HTTP headers that do not exceed 100 bytes. If a header is too large, the FWSM resets the TCP connection and creates a syslog entry.

```
hostname(config)# http-map inbound_http
hostname(config-http-map)# max-header-length request bytes 100 action log reset
hostname(config-http-map)# exit
```

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.

max-uri-length

To restrict HTTP traffic based on the length of the URI in the HTTP request message, use the **max-uri-length** command in HTTP map configuration mode, which is accessible using the **http-map** command. To remove this command, use the **no** form of this command.

max-uri-length *bytes* **action** { **allow** | **reset** | **drop** } [**log**]

no max-uri-length *bytes* **action** { **allow** | **reset** | **drop** } [**log**]

Syntax Description

action	The action taken when a message fails this command inspection.
allow	Allow the message.
drop	Closes the connection.
bytes	Number of bytes, range is 1 to 65535.
log	(Optional) Generate a syslog.
reset	Send a TCP reset message to client and server.

Defaults

This command is disabled 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
HTTP map configuration	•	•	•	•	—

Command History

Release	Modification
3.1	This command was introduced.

Usage Guidelines

After enabling the **max-uri-length** command, the FWSM only allows messages having a URI within the configured limit and otherwise takes the specified action. Use the **action** keyword to cause the FWSM to reset the TCP connection and create a syslog entry.

URIs with a length less than or equal to the configured value will be allowed. Otherwise, the specified action will be taken.

Examples

The following example restricts HTTP requests to those with URIs that do not exceed 100 bytes. If a URI is too large, the FWSM resets the TCP connection and creates a syslog entry.

```
hostname(config)# http-map inbound_http
hostname(config-http-map)# max-uri-length 100 action reset log
hostname(config-http-map)# exit
```

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.

mcc

To identify the mobile country code and the mobile network code for IMSI prefix filtering, use the **mcc** command in GTP map configuration mode. To remove the configuration, use the **no** form of this command.

```
mcc country_code mnc network_code
```

```
no mcc country_code mnc network_code
```

Syntax Description

<i>country_code</i>	A non-zero, three-digit value identifying the mobile country code. One or two-digit entries will be prepended by 0 to create a three-digit value.
<i>network_code</i>	A two or three-digit value identifying the network code.

Defaults

By default, the FWSM does not check for valid MCC/MNC combinations.

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
3.1(1)	This command was introduced.

Usage Guidelines

This command is used for IMSI Prefix filtering. The MCC and MNC in the IMSI of the received packet is compared with the MCC/MNC configured with this command and is dropped if it does not match.

This command must be used to enable IMSI Prefix filtering. You can configure multiple instances to specify permitted MCC and MNC combinations. By default, the FWSM does not check the validity of MNC and MCC combinations; therefore, you must verify the validity of the combinations configured. To find more information about MCC and MNC codes, see the ITU E.212 recommendation, *Identification Plan for Land Mobile Stations*.

Examples

The following example identifies traffic for IMSI Prefix filtering with an MCC of 111 and an MNC of 222:

```
hostname(config)# gtp-map gtp-policy
hostname(config-gtpmap)# mcc 111 mnc 222
```

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.

member

To assign a context to a resource class, use the **member** command in context configuration mode. To remove the context from the class, use the **no member** form of this command.

member *class_name*

no member *class_name*

Syntax Description	<i>class_name</i>	Specifies the class name you created with the class command.
--------------------	-------------------	---

Defaults	By default, the context is assigned to the default class.
----------	---

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
Context configuration	N/A	N/A	—	—	•

Command History	Release	Modification
	2.2(1)	This command was introduced.

Usage Guidelines	By default, all security contexts have unlimited access to the resources of the FWSM, except where maximum limits per context are enforced. However, if you find that one or more contexts use too many resources, and they cause other contexts to be denied connections, for example, then you can configure resource management to limit the use of resources per context. The FWSM manages resources by assigning contexts to resource classes. Each context uses the resource limits set by the class.
------------------	---

Examples	The following example assigns the context test to the gold class:
----------	---

```
hostname(config)# context test
hostname(config-ctx)# allocate-interface vlan100 int1
hostname(config-ctx)# allocate-interface vlan102 int2
hostname(config-ctx)# allocate-interface vlan110-vlan115 int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
hostname(config-ctx)# member gold
hostname(config-ctx)# allocate-acl-partition 0
```

Related Commands

Command	Description
class	Creates a resource class.
context	Configures a security context.
limit-resource	Sets the limit for a resource.
show resource allocation	Shows how you allocated resources across classes.
show resource types	Shows the resource types for which you can set limits.

memory caller-address

To configure a specific range of program memory for the call tracing, or caller PC, to help isolate memory problems, use the **memory caller-address** command in privileged EXEC mode. The caller PC is the address of the program that called a memory allocation primitive. To remove an address range, use the **no** form of this command.

memory caller-address *startPC endPC*

no memory caller-address

Syntax Description

<i>endPC</i>	Specifies the end address range of the memory block.
<i>startPC</i>	Specifies the start address range of the memory block.

Defaults

The actual caller PC is recorded for memory tracing.

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
3.1(1)	Support for this command was introduced.

Usage Guidelines

Use the **memory caller-address** command to isolate memory problems to a specific block of memory. In certain cases the actual caller PC of the memory allocation primitive is a known library function that is used at many places in the program. To isolate individual places in the program, configure the start and end program address of the library function, thereby recording the program address of the caller of the library function.



Note

The FWSM might experience a temporary reduction in performance when caller-address tracing is enabled.

Examples

The following examples show the address ranges configured with the **memory caller-address** commands, and the resulting display of the **show memory-caller address** command:

```
hostname# memory caller-address 0x00109d5c 0x00109e08
hostname# memory caller-address 0x009b0ef0 0x009b0f14
hostname# memory caller-address 0x00cf211c 0x00cf4464
```

```

hostname# show memory-caller address
Move down stack frame for the addresses:
pc = 0x00109d5c-0x00109e08
pc = 0x009b0ef0-0x009b0f14
pc = 0x00cf211c-0x00cf4464

```

Related Commands

Command	Description
memory profile enable	Enables the monitoring of memory usage (memory profiling).
memory profile text	Configures a text range of memory to profile.
show memory	Displays a summary of the maximum physical memory and current free memory available to the operating system.
show memory binsize	Displays summary information about the chunks allocated for a specific bin size.
show memory profile	Displays information about the memory usage (profiling) of the FWSM.
show memory-caller address	Displays the address ranges configured on the FWSM.

memory delayed-free-poisoner enable

To enable the delayed free-memory poisoner tool, use the **memory delayed-free-poisoner enable** command in privileged EXEC mode. To disable the delayed free-memory poisoner tool, use the **no** form of this command. The delayed free-memory poisoner tool lets you monitor freed memory for changes after it has been released by an application.

memory delayed free poisoner enable

no memory delayed free poisoner enable

Syntax Description This command has no arguments or keywords.

Defaults The **memory delayed-free-poisoner enable** command is disabled 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
Privileged EXEC	•	•	•	—	•

Command History	Release	Modification
	3.1(1)	This command was introduced.

Usage Guidelines Enabling the delayed free-memory poisoner tool has a significant impact on memory usage and system performance. The command should only be used under the supervision of the Cisco TAC. It should not be run in a production environment during heavy system usage.

When you enable this tool, requests to free memory by the applications running on the FWSM are written to a FIFO queue. As each request is written to the queue, each associated byte of memory that is not required by lower-level memory management is “poisoned” by being written with the value 0xcc.

The freed memory requests remain in the queue until more memory is required by an application than is in the free memory pool. When memory is needed, the first freed memory request is pulled from the queue and the poisoned memory is validated.

If the memory is unmodified, it is returned to the lower-level memory pool and the tool reissues the memory request from the application that made the initial request. The process continues until enough memory for the requesting application is freed.

If the poisoned memory has been modified, then the system forces a crash and produces diagnostic output to determine the cause of the crash.

The delayed free-memory poisoner tool periodically performs validation on all of the elements of the queue automatically. Validation can also be started manually using the **memory delayed-free-poisoner validate** command.

The **no** form of the command causes all of the memory referenced by the requests in the queue to be returned to the free memory pool without validation and any statistical counters to be cleared.

Examples

The following example enables the delayed free-memory poisoner tool:

```
hostname# memory delayed-free-poisoner
```

The following is sample output when the delayed free-memory poisoner tool detects illegal memory reuse:

```
delayed-free-poisoner validate failed because a
    data signature is invalid at delayfree.c:328.

    heap region:      0x025b1cac-0x025b1d63 (184 bytes)
    memory address:  0x025b1cb4
    byte offset:     8
    allocated by:    0x0060b812
    freed by:        0x0060ae15

Dumping 80 bytes of memory from 0x025b1c88 to 0x025b1cd7
025b1c80:                ef cd 1c a1 e1 00 00 00 | .....
025b1c90: 23 01 1c a1 b8 00 00 00 15 ae 60 00 68 ba 5e 02 | #.....`.h.^
025b1ca0: 88 1f 5b 02 12 b8 60 00 00 00 00 6c 26 5b 02 | ..[...`.l&[.
025b1cb0: 8e a5 ea 10 ff ff ff ff cc cc cc cc cc cc cc | .....
025b1cc0: cc cc cc cc cc cc cc cc cc cc cc cc cc cc cc | .....
025b1cd0: cc cc cc cc cc cc cc cc | .....
```

An internal error occurred. Specifically, a programming assertion was violated. Copy the error message exactly as it appears, and get the output of the show version command and the contents of the configuration file. Then call your technical support representative.

```
assertion "0" failed: file "delayfree.c", line 191
```

Table 20-1 describes the significant portion of the output.

Table 20-1 *Illegal Memory Usage Output Description*

Field	Description
heap region	The address region and size of the region of memory available for use by the requesting application. This is not the same as the requested size, which may be smaller given the manner in which the system may parcel out memory at the time the memory request was made.
memory address	The location in memory where the fault was detected.
byte offset	The byte offset is relative to the beginning of the heap region and can be used to find the field that was modified if the result was used to hold a data structure starting at this address. A value of 0 or that is larger than the heap region byte count may indicate that the problem is an unexpected value in the lower level heap package.

Table 20-1 *Illegal Memory Usage Output Description*

Field	Description
allocated by/freed by	Instruction addresses where the last malloc/calloc/realloc and free calls were made involving this particular region of memory.
Dumping...	A dump of one or two regions of memory, depending upon how close the detected fault was to the beginning of the region of heap memory. The next eight bytes after any system heap header is the memory used by this tool to hold a hash of various system header values plus the queue linkage. All other bytes in the region until any system heap trailer is encountered should be set to 0xcc.

Related Commands

Command	Description
clear memory delayed-free-poisoner	Clears the delayed free-memory poisoner tool queue and statistics.
memory delayed-free-poisoner validate	Forces validation of the elements in the delayed free-memory poisoner tool queue.
show memory delayed-free-poisoner	Displays a summary of the delayed free-memory poisoner tool queue usage.

memory delayed-free-poisoner validate

To force validation of all elements in the **memory delayed-free-poisoner** queue, use the **memory delayed-free-poisoner validate** command in privileged EXEC mode.

memory delayed free poisoner enable

Syntax Description

This command has no arguments or keywords.

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
Privileged EXEC	•	•	•	—	•

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

You must enable the delayed free-memory poisoner tool using the **memory delayed-free-poisoner enable** command before issuing the **memory delayed-free-poisoner validate** command.

The **memory delayed-free-poisoner validate** command causes each element of the **memory delayed-free-poisoner** queue to be validated. If an element contains unexpected values, then the system forces a crash and produces diagnostic output to determine the cause of the crash. If no unexpected values are encountered, the elements remain in the queue and are processed normally by the tool; the **memory delayed-free-poisoner validate** command does not cause the memory in the queue to be returned to the system memory pool.



Note

The delayed free-memory poisoner tool periodically performs validation on all of the elements of the queue automatically.

Examples

The following example causes all elements in the **memory delayed-free-poisoner** queue to be validated:

```
hostname# memory delayed-free-poisoner validate
```

Related Commands	Command	Description
	clear memory delayed-free-poisoner	Clears the delayed free-memory poisoner tool queue and statistics.
	memory delayed-free-poisoner enable	Enables the delayed free-memory poisoner tool.
	show memory delayed-free-poisoner	Displays a summary of the delayed free-memory poisoner tool queue usage.

memory profile enable

To enable the monitoring of memory usage (memory profiling), use the **memory profile enable** command in privileged EXEC mode. To disable memory profiling, use the **no** form of this command.

memory profile enable peak *peak_value*

no memory profile enable peak *peak_value*

Syntax Description

peak_value Specifies the memory usage threshold at which a snapshot of the memory usage is saved to the peak usage buffer. The contents of this buffer could be analyzed at a later time to determine the peak memory needs of the system.

Defaults

Memory profiling is disabled 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
Privileged EXEC	•	•	—	•	•

Command History

Release	Modification
3.1(1)	Support for this command was introduced.

Usage Guidelines

Before enabling memory profiling, you must first configure a memory text range to profile with the **memory profile text** command.

Some memory is held by the profiling system until you enter the **clear memory profile** command. See the output of the **show memory status** command.



Note

The FWSM might experience a temporary reduction in performance when memory profiling is enabled.

The following example enables memory profiling:

```
hostname# memory profile enable
```

Related Commands

Command	Description
memory profile text	Configures a text range of memory to profile.
show memory profile	Displays information about the memory usage (profiling) of the FWSM.

memory profile text

To configure a program text range of memory to profile, use the **memory profile text** command in privileged EXEC mode. To disable, use the **no** form of this command.

memory profile text {*startPC endPC* | **all** *resolution*}

no memory profile text {*startPC endPC* | **all** *resolution*}

Syntax Description

all	Specifies the entire text range of the memory block.
<i>endPC</i>	Specifies the end text range of the memory block.
<i>resolution</i>	Specifies the resolution of tracing for the source text region.
<i>startPC</i>	Specifies the start text range of the memory block.

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
Privileged EXEC	•	•	—	•	•

Command History

Release	Modification
3.1(1)	Support for this command was introduced.

Usage Guidelines

For a small text range, a resolution of “4” normally traces the call to an instruction. For a larger text range, a coarse resolution is probably enough for the first pass and the range could be narrowed down to a set of smaller regions in the next pass.

After entering the text range with the **memory profile text** command, you must then enter the **memory profile enable** command to begin memory profiling. Memory profiling is disabled by default.



Note

The FWSM might experience a temporary reduction in performance when memory profiling is enabled.

Examples

The following example shows how to configure a text range of memory to profile, with a resolution of 4:

```
hostname# memory profile text 0x004018b4 0x004169d0 4
```

The following example displays the configuration of the text range and the status of memory profiling (OFF):

```
hostname# show memory profile
InUse profiling: OFF
Peak profiling: OFF
Profile:
0x004018b4-0x004169d0(00000004)
```

**Note**

To begin memory profiling, you must enter the **memory profile enable** command. Memory profiling is disabled by default.

Related Commands

Command	Description
clear memory profile	Clears the buffers held by the memory profiling function.
memory profile enable	Enables the monitoring of memory usage (memory profiling).
show memory profile	Displays information about the memory usage (profiling) of the FWSM.
show memory-caller address	Displays the address ranges configured on the FWSM.

message-length

To filter GTP packets that do not meet the configured maximum and minimum length, use the **message-length** command in GTP map configuration mode, which is accessed by using the **gtp-map** command. Use the **no** form to remove the command.

```
message-length min min_bytes max max_bytes
```

```
no message-length min min_bytes max max_bytes
```

Syntax Description

max	Specifies the maximum number of bytes allowed in the UDP payload.
<i>max_bytes</i>	The maximum number of bytes in the UDP payload. The range is from 1 to 65536.
min	Specifies the minimum number of bytes allowed in the UDP payload.
<i>min_bytes</i>	The minimum number of bytes in the UDP payload. The range is from 1 to 65536.

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
GTP map configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The length specified by this command is the sum of the GTP header and the rest of the message, which is the payload of the UDP packet.

Examples

The following example allows messages between 20 bytes and 300 bytes in length:

```
hostname(config)# gtp-map gtp-policy
hostname(config-gtpmap)# permit message-length min 20 max 300
```

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.

mfib forwarding

To reenable MFIB forwarding on an interface, use the **mfib forwarding** command in interface configuration mode. To disable MFIB forwarding on an interface, use the **no** form of this command.

mfib forwarding

no mfib forwarding

Syntax Description

This command has no arguments or keywords.

Defaults

The **multicast-routing** command enables MFIB forwarding on all interfaces 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
Interface configuration	•	—	•	—	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

When you enable multicast routing, MFIB forwarding is enabled on all interfaces by default. Use the **no** form of the command to disable MFIB forwarding on a specific interface. Only the **no** form of the command appears in the running configuration.

When MFIB forwarding is disabled on an interface, the interface does not accept any multicast packets unless specifically configured through other methods. IGMP packets are also prevented when MFIB forwarding is disabled.

Examples

The following example disables MFIB forwarding on the specified interface:

```
hostname(config)# interface Vlan55
hostname(config-if)# no mfib forwarding
```

Related Commands

Command	Description
multicast-routing	Enables multicast routing.
pim	Enables PIM on an interface.

mgcp-map

To identify a specific map for defining the parameters for MGCP inspection, use the **mgcp-map** command in global configuration mode. To remove the map, use the **no** form of this command.

```
mgcp-map map_name
```

```
no mgcp-map map_name
```

Syntax Description

<i>map_name</i>	The name of the MGCP map. The maximum number of characters is 64.
-----------------	---

Defaults

The default for the MGCP command queue 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
Global configuration	•	•	•	•	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Use the **mgcp-map** command to identify a specific map to use for defining the parameters for MGCP inspection. When you enter this command, the system enters a configuration mode that lets you enter the different commands used for defining the specific map. After defining the MGCP map, you use the **inspect mgcp** command to enable the map. You use Modular Policy Framework to apply the **inspect** command to a defined class of traffic and to apply the policy to a specific interface. The following are the commands available in MGCP map configuration mode.

- **call-agent**—Specifies a group of call agents.
- **command-queue**—Specifies the maximum number of MGCP commands that can be queued.
- **gateway**—Specifies the group of call agents that are managing a particular gateway.
- **no**—Negates a command or sets a parameter to its default value.

Examples

The following example shows how to use the **mgcp-map** command to identify a specific map (mgcp-policy) to use for defining the parameters for MGCP inspection.

```
hostname(config)# mgcp-map mgcp-policy
hostname(config-mgcp-policy)#
```

The following example shows how to identify MGCP traffic, define a MGCP map, define a policy, and apply the policy to the outside interface. You enable the MGCP inspection engine as shown in the following example, which creates a class map to match MGCP traffic on the default port (2427). The service policy is then applied to the outside interface.

```
hostname(config)# class-map mgcp-port
hostname(config-cmap)# match port tcp eq 2427
hostname(config-cmap)# exit
hostname(config)# mgcp-map mgcp_inbound
hostname(config-mgcp-map)# call-agent 10.10.11.5 101
hostname(config-mgcp-map)# call-agent 10.10.11.6 101
hostname(config-mgcp-map)# call-agent 10.10.11.7 102
hostname(config-mgcp-map)# call-agent 10.10.11.8 102
hostname(config-mgcp-map)# gateway 10.10.10.115 101
hostname(config-mgcp-map)# gateway 10.10.10.116 102
hostname(config-mgcp-map)# gateway 10.10.10.117 102
hostname(config-mgcp-map)# command-queue 150
hostname(config)# policy-map mgcp_policy
hostname(config-pmap)# class mgcp-port
hostname(config-pmap-c)# inspect mgcp mgcp_inbound
hostname(config-pmap-c)# exit
hostname(config)# service-policy mgcp_policy interface outside
```

This allows call agents 10.10.11.5 and 10.10.11.6 to control gateway 10.10.10.115, and allows call agents 10.10.11.7 and 10.10.11.8 to control both gateways 10.10.10.116 and 10.10.10.117. The maximum number of MGCP commands that can be queued is 150.

To enable MGCP inspection for all interfaces, use the **global** parameter in place of **interface outside**.

Related Commands

Commands	Description
debug mgcp	Enables the display of debug information for MGCP.
show mgcp	Displays MGCP configuration and session information.
timeout	Configures the idle timeouts related to MGCP.

mkdir

To create a new directory, use the **mkdir** command in privileged EXEC mode.

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

Syntax Description

noconfirm	(Optional) Suppresses the confirmation prompt.
flash:	(Optional) Specifies the internal Flash memory, followed by a colon.
<i>path</i>	The name and path of the directory to create.

Defaults

If you do not specify a path, the directory is created in the current working directory.

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
3.1(1)	Support for this command was introduced.

Usage Guidelines

If a directory with the same name already exists, then the new directory is not created.

Examples

This example shows how to make a new directory called “backup”:

```
hostname# mkdir backup
```

Related Commands

Command	Description
cd	Changes the current working directory to the one specified.
dir	Displays the directory contents.
rmdir	Removes the specified directory.
pwd	Display the current working directory.

mode

To set the security context mode to single or multiple, use the **mode** command in global configuration mode. You can partition a single FWSM into multiple virtual devices, known as security contexts. Each context behaves like an independent device, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple standalone appliances. In single mode, the FWSM has a single configuration and behaves as a single device. In multiple mode, you can create multiple contexts, each with its own configuration. The number of contexts allowed depends on your license.

mode { **single** | **multiple** } [**noconfirm**]

Syntax Description

multiple	Sets multiple context mode.
noconfirm	(Optional) Sets the mode without prompting you for confirmation. This option is useful for automated scripts.
single	Sets the context mode to single.

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
2.2(1)	This command was introduced.

Usage Guidelines

In multiple context mode, the FWSM includes a configuration for each context that identifies the security policy, interfaces, and almost all the options you can configure on a stand-alone device (see the **config-url** command to identify the context configuration location). The system administrator adds and manages contexts by configuring them in the system configuration, which, like a single mode configuration, is the startup configuration. The system configuration identifies basic settings for the FWSM. The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the admin context.

When you change the context mode using the **mode** command, you are prompted to reboot.

The context mode (single or multiple) is not stored in the configuration file, even though it does endure reboots. If you need to copy your configuration to another device, set the mode on the new device to match using the **mode** command.

When you convert from single mode to multiple mode, the FWSM converts the running configuration into two files: a new startup configuration that comprises the system configuration, and admin.cfg that comprises the admin context (in the root directory of the internal Flash memory). The original running configuration is saved as old_running.cfg (in the root directory of the internal Flash memory). The original startup configuration is not saved. The FWSM automatically adds an entry for the admin context to the system configuration with the name “admin.”

If you convert from multiple mode to single mode, you might want to first copy a full startup configuration (if available) to the FWSM; the system configuration inherited from multiple mode is not a complete functioning configuration for a single mode device.

Not all features are supported in multiple context mode. See the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Configuration Guide* for more information.

Examples

The following example sets the mode to multiple:

```
hostname(config)# mode multiple
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm] y
Convert the system configuration? [confirm] y
Flash Firewall mode: multiple

***
*** --- SHUTDOWN NOW ---
***
*** Message to all terminals:
***
*** change mode

Rebooting...

Booting system, please wait...
```

The following example sets the mode to single:

```
hostname(config)# mode single
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm] y
Flash Firewall mode: single

***
*** --- SHUTDOWN NOW ---
***
*** Message to all terminals:
***
*** change mode

Rebooting...

Booting system, please wait...
```

Related Commands	Command	Description
	context	Configures a context in the system configuration and enters context configuration mode.
	show mode	Shows the current context mode, either single or multiple.

monitor-interface

To enable health monitoring on a specific interface, use the **monitor-interface** command in global configuration mode. To disable interface monitoring, use the **no** form of this command.

monitor-interface *if_name*

no monitor-interface *if_name*

Syntax Description

if_name Specifies the name of the interface being monitored.

Defaults

Monitoring of logical interfaces is disabled 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
Global configuration	•	•	•	•	—

Command History

Release	Modification
2.2(1)	This command was introduced.

Usage Guidelines

The number of interfaces that can be monitored for the FWSM is 250. Hello messages are exchanged during every interface poll frequency time period between the FWSM failover pair. The failover interface poll time is 3 to 15 seconds. For example, if the poll time is set to 5 seconds, testing begins on an interface if 5 consecutive hellos are not heard on that interface (25 seconds).

Monitored failover interfaces can have the following status:

- Unknown—Initial status. This status can also mean the status cannot be determined.
- Normal—The interface is receiving traffic.
- Testing—Hello messages are not heard on the interface for five poll times.
- Link Down—The interface or VLAN is administratively down.
- No Link—The physical link for the interface is down.
- Failed—No traffic is received on the interface, yet traffic is heard on the peer interface.

In Active/Active failover, this command is only valid within a context.

Examples

The following example enables monitoring on an interface named “inside”:

```
hostname(config)# monitor-interface inside
```

■ monitor-interface

```
hostname(config)#
```

Related Commands	Command	Description
	clear configure monitor-interface	Removes the monitor-interface commands from the running configuration.
	failover interface-policy	Specifies the number or percentage of monitored interface that must fail for failover to occur.
	failover polltime	Specifies the interval between hello messages on an interface (Active/Standby failover).
	polltime interface	Specifies the interval between hello messages on an interface (Active/Active failover).
	show running-config monitor-interface	Displays the monitor-interface commands in the running configuration.

more

To display the contents of a file, use the **more** command in privileged EXEC mode.

more {/ascii | /binary | /ebcdic | flash: | ftp: | http: | https: | system: | tftp:}filename

Syntax Description	
/ascii	(Optional) Displays a binary file in binary mode and an ASCII file in binary mode.
/binary	(Optional) Displays any file in binary mode.
/ebcdic	(Optional) Displays binary files in EBCDIC.
flash:	(Optional) Specifies the internal Flash memory, followed by a colon.
ftp:	(Optional) Displays a file on an FTP server.
http:	(Optional) Displays a file on a web site.
https:	(Optional) Displays a file on a secure web site.
system:	(Optional) Displays the file system.
tftp:	(Optional) Displays a file on a TFTP server.
filename	Specifies the name of the file to display.

Defaults

ASCII mode

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
2.2(1)	This command was introduced.

Usage Guidelines

The **more filesystem:** command prompts you to enter the alias of the local directory or file systems.

Examples

This example shows how to display the contents of a local file named “test.cfg”:

```
hostname# more test.cfg
: Saved
: Written by enable_15 at 10:04:01 Apr 14 2005

XXX Version X.X(X)
nameif vlan300 outside security10
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
```

more

```

hostname test
fixup protocol ftp 21
fixup protocol h323 H225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
access-list deny-flow-max 4096
access-list alert-interval 300
access-list 100 extended permit icmp any any
access-list 100 extended permit ip any any
pager lines 24
icmp permit any outside
mtu outside 1500
ip address outside 172.29.145.35 255.255.0.0
no asdm history enable
arp timeout 14400
access-group 100 in interface outside
!
interface outside
!
route outside 0.0.0.0 0.0.0.0 172.29.145.1 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02 rpc 0:10:00 h3
23 0:05:00 h225 1:00:00 mgcp 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
snmp-server host outside 128.107.128.179
snmp-server location my_context, USA
snmp-server contact admin@my_context.com
snmp-server community public
no snmp-server enable traps
floodguard enable
fragment size 200 outside
no sysopt route dnat
telnet timeout 5
ssh timeout 5
terminal width 511
gdb enable
mgcp command-queue 0
Cryptochecksum:00000000000000000000000000000000
: end

```

Related Commands

Command	Description
cd	Changes to the specified directory.
pwd	Displays the current working directory.

mroute

To configure a static multicast route, use the **mroute** command in global configuration mode. To remove a static multicast route, use the **no** form of this command.

```
mroute src smask {in_if_name | rpf_neighbor} [dense output_if_name] [distance]
```

```
no mroute src smask {in_if_name | rpf_neighbor} [dense output_if_name] [distance]
```

Syntax Description

<i>dense output_if_name</i>	(Optional) The interface name for dense mode output. The dense output_if_name keyword and argument pair is only supported for SMR stub multicast routing (igmp forwarding).
<i>distance</i>	(Optional) The administrative distance of the route. Routes with lower distances have preference. The default is 0.
<i>in_if_name</i>	Specifies the incoming interface name for the mroute.
<i>rpf_neighbor</i>	Specifies the RPF neighbor for the security appliance.
<i>smask</i>	Specifies the multicast source network address mask.
<i>src</i>	Specifies the IP address of the multicast source.

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
3.1(1)	This command was introduced.

Usage Guidelines

This command lets you statically configure where multicast sources are located. The FWSM expects to receive multicast packets on the same interface as it would use to send unicast packets to a specific source. In some cases, such as bypassing a route that does not support multicast routing, multicast packets may take a different path than the unicast packets.

Static multicast routes are not advertised or redistributed.



Note

You can specify the interface name or the RPF neighbor using this command, but not at the same time.

Use the **show mroute** command displays the contents of the multicast route table. Use the **show running-config mroute** command to display the mroute commands in the running configuration.

Examples

The following example shows how configure a static multicast route using the **mroute** command:

```
hostname(config)# mroute 172.16.0.0 255.255.0.0 inside
```

Related Commands

Command	Description
show running-config mroute	Displays the mroute commands in the configuration.

mtu

To specify the maximum transmission unit for an interface, use the **mtu** command in global configuration mode. To reset the MTU block size to 1500 for Ethernet interfaces, use the **no** form of this command. This command supports IPv4 and IPv6 traffic.

mtu *interface_name* *bytes*

no mtu *interface_name* *bytes*

Syntax Description

<i>bytes</i>	Number of bytes in the MTU; valid values are from 64 to 65,535 bytes.
<i>interface_name</i>	Internal or external network interface name.

Defaults

The default *bytes* is 1500 for Ethernet interfaces.

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
1.1(1)	This command was introduced.

Usage Guidelines

The **mtu** command lets you to set the data size that is sent on a connection. Data that is larger than the MTU value is fragmented before being sent.

The FWSM supports IP path MTU discovery (as defined in RFC 1191), which allows a host to dynamically discover and cope with the differences in the maximum allowable MTU size of the various links along the path. Sometimes, the FWSM cannot forward a datagram because the packet is larger than the MTU that you set for the interface, but the “don’t fragment” (DF) bit is set. The network software sends a message to the sending host, alerting it to the problem. The host has to fragment packets for the destination so that they fit the smallest packet size of all the links along the path.

The default MTU is 1500 bytes in a block for Ethernet interfaces (which is also the maximum). This value is sufficient for most applications, but you can pick a lower number if network conditions require it.

When using the Layer 2 Tunneling Protocol (L2TP), we recommend that you set the MTU size to 1380 to account for the L2TP header and IPSec header length.

Examples

This example shows how to specify the MTU for an interface:

```
hostname(config)# show running-config mtu
mtu outside 1500
mtu inside 1500
hostname(config)# mtu inside 8192
hostname(config)# show running-config mtu
mtu outside 1500
mtu inside 8192
```

Related Commands

Command	Description
clear configure mtu	Clears the configured maximum transmission unit values on all interfaces.
show running-config mtu	Displays the current maximum transmission unit block size.

multicast-routing

To enable IP multicast routing on the FWSM, use the **multicast routing** command in global configuration mode. To disable IP multicast routing, use the **no** form of this command.

multicast-routing

no multicast-routing

Syntax Description

This command has no arguments or keywords.

Defaults

The **multicast-routing** command enables PIM and IGMP on all interfaces 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
Global configuration	•	—	•	—	—

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The **multicast-routing** command enables PIM and IGMP on all interfaces.



Note

PIM is not supported with PAT. The PIM protocol does not use ports and PAT only works with protocols that use ports.

If the security appliance is the PIM RP, use the untranslated outside address of the security appliance as the RP address.

The number of entries in the multicast routing tables are limited by the amount of RAM on the system. [Table 20-2](#) lists the maximum number of entries for specific multicast tables based on the amount of RAM on the security appliance. Once these limits are reached, any new entries are discarded.

Table 20-2 *Entry Limits for Multicast Tables*

Table	16 MB	128 MB	128+ MB
MFIB	1000	3000	5000
IGMP Groups	1000	3000	5000
PIM Routes	3000	7000	12000

Examples

The following example enables IP multicast routing on the FWSM:

```
hostname(config)# multicast-routing
```

Related Commands

Command	Description
igmp	Enables IGMP on an interface.
pim	Enables PIM on an interface.