maa – match d

mac address

To specify the virtual MAC addresses for the active and standby units, use the **mac address** command in failover group configuration mode. To restore the default virtual MAC addresses, use the **no** form of this command.

mac address *phy_if* [*active_mac*] [*standby_mac*]
no mac address *phy_if* [*active_mac*] [*standby_mac*]

Syntax Description

phy_if	The physical name of the interface to set the MAC address.
active_mac	The virtual MAC address for the active unit. The MAC address must be entered in h.h.h format, where h is a 16-bit hexadecimal number.
standby_mac	The virtual MAC address for the standby unit. The MAC address must be entered in h.h.h format, where h is a 16-bit hexadecimal number.

Command Default

The defaults are as follows:

- Active unit default MAC address: 00a0.c9*physical_port_number .failover_group_id* 01.
- Standby unit default MAC address: 00a0.c9*physical_port_number .failover_group_id* 02.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

If the virtual MAC addresses are not defined for the failover group, the default values are used.

If you have more than one Active/Active failover pair on the same network, it is possible to have the same default virtual MAC addresses assigned to the interfaces on one pair as are assigned to the interfaces of the other pairs because of the way the default virtual MAC addresses are determined. To avoid having duplicate MAC addresses on your network, make sure you assign each physical interface a virtual active and standby MAC address.

You can also set the MAC address using other commands or methods, but we recommend using only one method. If you set the MAC address using multiple methods, the MAC address used depends on many variables, and might not be predictable.

Examples

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

```
ciscoasa(config)# failover group 1

ciscoasa(config-fover-group)# primary
ciscoasa(config-fover-group)# preempt 100
ciscoasa(config-fover-group)# exit
ciscoasa(config)# failover group 2
ciscoasa(config-fover-group)# secondary
ciscoasa(config-fover-group)# preempt 100
ciscoasa(config-fover-group)# mac address e1 0000.a000.a011 0000.a000.a012

ciscoasa(config-fover-group)# exit
ciscoasa(config)#
```

Related Commands

Command	Description
failover group	Defines a failover group for Active/Active failover.
failover mac address	Specifies a virtual MAC address for a physical interface.

mac-address

To manually assign a private MAC address to an interface or subinterface, use the **mac-address** command in interface configuration mode. In multiple context mode, this command can assign a different MAC address to the interface in each context. For an individual interface in a cluster, you can assign a cluster pool of MAC addresses. To revert the MAC address to the default, use the **no** form of this command.

mac-address { *mac_address* [**standby** *mac_address* | **site-id** *number* [**site-ip ip_address**]] | **cluster-pool** *pool_name* }
no mac-address { *mac_address* [**standby** *mac_address* | **site-id** *number* [**site-ip ip_address**]] | **cluster-pool** *pool_name* }

Syntax Description

cluster-pool *pool_name*	For a cluster in individual interface mode (see the **cluster interface-mode** command), or for a management interface in any cluster interface mode, sets a pool of MAC addresses to be used for a given interface on each cluster member. Define the pool using the **mac-address pool** command.
mac_address	Sets the MAC address for this interface in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE would be entered as 000C.F142.4CDE. If you use failover, this MAC address is the active MAC address. **Note** Because auto-generated addresses (the **mac-address auto** command) start with A2, you cannot start manual MAC addresses with A2 if you also want to use auto-generation.
site-id *number*	(Optional; Routed mode only) For inter-site clustering, configures a site-specific MAC address for each site.
site-ip *ip_address*	(Optional; Routed mode only) For inter-site clustering, configures a site-specific IP address for each site. The IP address must be on the same subnet as the global IP address.
standby *mac_address*	(Optional) Sets the standby MAC address for failover. If the active unit fails over and the standby unit becomes active, the new active unit starts using the active MAC addresses to minimize network disruption, while the old active unit uses the standby address.

Command Default

The default MAC address is the burned-in MAC address of the physical interface. Subinterfaces inherit the physical interface MAC address. Some commands set the physical interface MAC address (including this command in single mode), so the inherited address depends on that configuration.

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.
8.0(5)/8.2(2)	The use of A2 to start the MAC address was restricted when also used with the **mac-address auto** command.
9.0(1)	The **cluster-pool** keyword was added to support clustering.
9.5(1)	The **site-id** keyword was added.
9.6(1)	The **site-ip** keyword was added.

Usage Guidelines

In multiple context mode, if you share an interface between contexts, you can assign a unique MAC address to the interface in each context. This feature lets the ASA easily classify packets into the appropriate context. Using a shared interface without unique MAC addresses is possible, but has some limitations. See the CLI configuration guide for more information.

You can assign each MAC address manually with this command, or you can automatically generate MAC addresses for shared interfaces in contexts using the **mac-address auto** command. If you automatically generate MAC addresses, you can use the **mac-address** command to override the generated address.

For single context mode, or for interfaces that are not shared in multiple context mode, you might want to assign unique MAC addresses to subinterfaces. For example, your service provider might perform access control based on the MAC address.

You can also set the MAC address using other commands or methods, but we recommend using only one method. If you set the MAC address using multiple methods, the MAC address used depends on many variables, and might not be predictable.

For clustering, you must configure a global MAC address for a Spanned EtherChannel. With a manually-configured MAC address, the MAC address stays with the current master unit. In multiple context mode, if you share an interface between contexts, you should enable auto-generation of MAC addresses. Note that you must manually configure the MAC address for non-shared interfaces.

For inter-site clustering in routed mode, configure a site-specific MAC address and IP address on the master unit for each site, then use the **site-id** command on each unit to assign it to a site.

Examples

The following example configures the MAC address for GigabitEthernet 0/1.1:

```
ciscoasa/contextA(config)# interface gigabitethernet0/1.1
ciscoasa/contextA(config-if)# nameif inside
ciscoasa/contextA(config-if)# security-level 100
ciscoasa/contextA(config-if)# ip address 10.1.2.1 255.255.255.0
```

```
ciscoasa/contextA(config-if)# mac-address 030C.F142.4CDE standby 040C.F142.4CDE
ciscoasa/contextA(config-if)# no shutdown
```

The following example configures site-specific MAC addresses for Spanned EtherChannel port-channel 1:

```
ciscoasa(config-if)# interface port-channel 1
ciscoasa(config-if)# port-channel span-cluster
ciscoasa(config-if)# mac-address aaaa.1111.1234
ciscoasa(config-if)# mac-address aaaa.1111.aaaa site-id 1 site-ip 10.7.7.1
ciscoasa(config-if)# mac-address aaaa.1111.bbbb site-id 2 site-ip 10.7.7.2
ciscoasa(config-if)# mac-address aaaa.1111.cccc site-id 3 site-ip 10.7.7.3
ciscoasa(config-if)# mac-address aaaa.1111.dddd site-id 4 site-ip 10.7.7.4
```

Related Commands

Command	Description
failover mac address	Sets the active and standby MAC address of a physical interface for Active/Standby failover.
mac address	Sets the active and standby MAC address of a physical interface for Active/Active failover.
mac-address auto	Auto-generates MAC addresses (active and standby) for shared interfaces in multiple context mode.
mode	Sets the security context mode to multiple or single.
show interface	Shows the interface characteristics, including the MAC address.

mac-address auto

To automatically assign private MAC addresses to each shared context interface, use the **mac-address auto** command in global configuration mode. To disable automatic MAC addresses, use the **no** form of this command.

mac-address auto [**prefix** *prefix*]
no mac-address auto

Syntax Description

prefix *prefix*	(Optional) Sets a user-defined prefix as part of the MAC address. The *prefix* is a decimal value between 0 and 65535. If you do not enter a prefix, then the ASA generates a default prefix. This prefix is converted to a 4-digit hexadecimal number. The prefix ensures that each ASA uses unique MAC addresses (using different prefix values), so you can have multiple ASAs on a network segment, for example.

Command Default

Automatic MAC address generation is disabled by default, except for the ASASM, where it is enabled by default. When enabled, the ASA autogenerates the prefix based on the last two bytes of the interface (ASA 5500-X) or backplane (ASASM) MAC address. You can customize the prefix if desired.

If you disable MAC address generation, see the following default MAC addresses:

- For the ASA 5500-X series appliances—The physical interface uses the burned-in MAC address, and all subinterfaces of a physical interface use the same burned-in MAC address.
- For the ASASM—All VLAN interfaces use the same MAC address, derived from the backplane MAC 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
Global configuration	• Yes	• Yes	—	—	• Yes

Command History

Release	Modification
7.2(1)	This command was added.
8.0(5)/8.2(2)	The **prefix** keyword was added. The MAC address format was changed to use the prefix, to use a fixed starting value (A2), and to use a different scheme for the primary and secondary unit MAC addresses in a failover pair. The MAC addresses are also now persistent across reloads. The command parser now checks if auto-generation is enabled; if you want to also manually assign a MAC address, you cannot start the manual MAC address with A2.
8.5(1)	Autogeneration is now enabled by default (**mac-address auto**) for the ASASM only.

Release	Modification
8.6(1)	The ASA now converts the automatic MAC address generation configuration to use a default prefix. The ASA auto-generates the prefix based on the last two bytes of the interface (ASA 5500) or backplane (ASASM) MAC address. This conversion happens automatically when you reload, or if you reenable MAC address generation. The legacy method of MAC address generation is no longer available. **Note** To maintain hitless upgrade for failover pairs, the ASA does *not* convert the MAC address method in an existing configuration upon a reload if failover is enabled.

Usage Guidelines

To allow contexts to share interfaces, we suggest that you assign unique MAC addresses to each shared context interface. The MAC address is used to classify packets within a context. If you share an interface, but do not have unique MAC addresses for the interface in each context, then the destination IP address is used to classify packets. The destination address is matched with the context NAT configuration, and this method has some limitations compared to the MAC address method. See the CLI configuration guide for information about classifying packets.

In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface within the context. See the **mac-address** command to manually set the MAC address.

Interaction with Manual MAC Addresses

If you manually assign a MAC address and also enable auto-generation, then the manually assigned MAC address is used. If you later remove the manual MAC address, the auto-generated address is used.

Because auto-generated addresses start with A2, you cannot start manual MAC addresses with A2 if you also want to use auto-generation.

Failover MAC Addresses

For use with failover, the ASA generates both an active and standby MAC address for each interface. If the active unit fails over and the standby unit becomes active, the new active unit starts using the active MAC addresses to minimize network disruption. See the <xref> section for more information.

For upgrading failover units with the legacy version of the **mac-address auto** command before the **prefix** keyword was added, see the <xref> section.

MAC Address Format Using a Prefix

The ASA generates the MAC address using the following format:

A2*xx.yyzz.zzzz*

Where *xx.yy* is a user-defined prefix or an autogenerated prefix based on the last two bytes of the interface (ASA 5500) or backplane (ASASM) MAC address, and *zz.zzzz* is an internal counter generated by the ASA. For the standby MAC address, the address is identical except that the internal counter is increased by 1.

For an example of how the prefix is used, if you set a prefix of 77, then the ASA converts 77 into the hexadecimal value 004D (*yyxx*). When used in the MAC address, the prefix is reversed (*xxyy*) to match the ASA native form:

A2**4D.00***zz.zzzz*

For a prefix of 1009 (03F1), the MAC address is:

A2**F1.03***zz.zzzz*

MAC Address Format Without a Prefix (Legacy Method)

This method may be used if you use failover and you upgraded to Version 8.6 or later; in this case, you have to manually enable the prefix method.

Without a prefix, the MAC address is generated using the following format:

- Active unit MAC address: 12_*slot .port _subid .contextid* .
- Standby unit MAC address: 02_*slot .port _subid .contextid* .

For platforms with no interface slots, the slot is always 0. The *port* is the interface port. The *subid* is an internal ID for the subinterface, which is not viewable. The *contextid* is an internal ID for the context, viewable with the **show context detail** command. For example, the interface GigabitEthernet 0/1.200 in the context with the ID 1 has the following generated MAC addresses, where the internal ID for subinterface 200 is 31:

- Active: 1200.0131.0001
- Standby: 0200.0131.0001

This MAC address generation method does not allow for persistent MAC addresses across reloads, does not allow for multiple ASAs on the same network segment (because unique MAC addresses are not guaranteed), and does not prevent overlapping MAC addresses with manually assigned MAC addresses. We recommend using a prefix with the MAC address generation to avoid these issues.

When the MAC Address is Generated

When you configure a **nameif** command for the interface in a context, the new MAC address is generated immediately. If you enable this command after you configure context interfaces, then MAC addresses are generated for all interfaces immediately after you enter the command. If you use the **no mac-address auto** command, the MAC address for each interface reverts to the default MAC address. For example, subinterfaces of GigabitEthernet 0/1 revert to using the MAC address of GigabitEthernet 0/1.

Setting the MAC Address Using Other Methods

You can also set the MAC address using other commands or methods, but we recommend using only one method. If you set the MAC address using multiple methods, the MAC address used depends on many variables, and might not be predictable.

Viewing MAC Addresses in the System Configuration

To view the assigned MAC addresses from the system execution space, enter the **show running-config all context** command.

The **all** option is required to view the assigned MAC addresses. Although this command is user-configurable in global configuration mode only, the **mac-address auto** command appears as a read-only entry in the configuration for each context along with the assigned MAC address. Only allocated interfaces that are configured with a **nameif** command within the context have a MAC address assigned.

Note If you manually assign a MAC address to an interface, but also have auto-generation enabled, the auto-generated address continues to show in the configuration even though the manual MAC address is the one that is in use. If you later remove the manual MAC address, the auto-generated one shown will be used.

Viewing MAC Addresses Within a Context

To view the MAC address in use by each interface within the context, enter the **show interface | include (Interface)|(MAC)** command.

> **Note** The **show interface** command shows the MAC address in use; if you manually assign a MAC address and also have auto-generation enabled, then you can only view the unused auto-generated address from within the system configuration.

Examples

The following example enables automatic MAC address generation with a prefix of 78:

```
ciscoasa(config)# mac-address auto prefix 78
```

The following output from the **show running-config all context admin** command shows the primary and standby MAC address assigned to the Management0/0 interface:

```
ciscoasa# show running-config all context admin
context admin
  allocate-interface Management0/0
  mac-address auto Management0/0 a24d.0000.1440 a24d.0000.1441
  config-url disk0:/admin.cfg
```

The following output from the **show running-config all context** command shows all the MAC addresses (primary and standby) for all context interfaces. Note that because the GigabitEthernet0/0 and GigabitEthernet0/1 main interfaces are not configured with a **nameif** command inside the contexts, no MAC addresses have been generated for them.

```
ciscoasa# show running-config all context
admin-context admin
context admin
  allocate-interface Management0/0
  mac-address auto Management0/0 a2d2.0400.125a a2d2.0400.125b
  config-url disk0:/admin.cfg
!
context CTX1
  allocate-interface GigabitEthernet0/0
  allocate-interface GigabitEthernet0/0.1-GigabitEthernet0/0.5
  mac-address auto GigabitEthernet0/0.1 a2d2.0400.11bc a2d2.0400.11bd
  mac-address auto GigabitEthernet0/0.2 a2d2.0400.11c0 a2d2.0400.11c1
  mac-address auto GigabitEthernet0/0.3 a2d2.0400.11c4 a2d2.0400.11c5
  mac-address auto GigabitEthernet0/0.4 a2d2.0400.11c8 a2d2.0400.11c9
  mac-address auto GigabitEthernet0/0.5 a2d2.0400.11cc a2d2.0400.11cd
  allocate-interface GigabitEthernet0/1
  allocate-interface GigabitEthernet0/1.1-GigabitEthernet0/1.3
  mac-address auto GigabitEthernet0/1.1 a2d2.0400.120c a2d2.0400.120d
  mac-address auto GigabitEthernet0/1.2 a2d2.0400.1210 a2d2.0400.1211
  mac-address auto GigabitEthernet0/1.3 a2d2.0400.1214 a2d2.0400.1215
  config-url disk0:/CTX1.cfg
!
context CTX2
  allocate-interface GigabitEthernet0/0
  allocate-interface GigabitEthernet0/0.1-GigabitEthernet0/0.5
  mac-address auto GigabitEthernet0/0.1 a2d2.0400.11ba a2d2.0400.11bb
  mac-address auto GigabitEthernet0/0.2 a2d2.0400.11be a2d2.0400.11bf
  mac-address auto GigabitEthernet0/0.3 a2d2.0400.11c2 a2d2.0400.11c3
  mac-address auto GigabitEthernet0/0.4 a2d2.0400.11c6 a2d2.0400.11c7
  mac-address auto GigabitEthernet0/0.5 a2d2.0400.11ca a2d2.0400.11cb
  allocate-interface GigabitEthernet0/1
  allocate-interface GigabitEthernet0/1.1-GigabitEthernet0/1.3
  mac-address auto GigabitEthernet0/1.1 a2d2.0400.120a a2d2.0400.120b
  mac-address auto GigabitEthernet0/1.2 a2d2.0400.120e a2d2.0400.120f
  mac-address auto GigabitEthernet0/1.3 a2d2.0400.1212 a2d2.0400.1213
  config-url disk0:/CTX2.cfg
!
```

Related Commands

Command	Description
failover mac address	Sets the active and standby MAC address of a physical interface for Active/Standby failover.
mac address	Sets the active and standby MAC address of a physical interface for Active/Active failover.
mac-address	Manually sets the MAC address (active and standby) for a physical interface or subinterface. In multiple context mode, you can set different MAC addresses in each context for the same interface.
mode	Sets the security context mode to multiple or single.
show interface	Shows the interface characteristics, including the MAC address.

mac-address pool

To add a MAC address pool for use on an individual interface in an ASA cluster, use the **mac-address pool** command in global configuration mode. To remove an unused pool, use the **no** form of this command.

mac-address pool *name start_mac_address* - *end_mac_address*
no mac-address pool *name* [*start_mac_address* - *end_mac_address*]

Syntax Description

name	Names the pool up to 63 characters in length.
start_mac_address - *end_mac_address*	Specifies the first MAC address and the last MAC address. Note to add a space around the dash (-).

Command Default

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	• Yes	• Yes	• Yes	—	• Yes

Command History

Release	Modification
9.0(1)	This command was added.

Usage Guidelines

You can use the pool in the **mac-address cluster-pool** command in interface configuration mode. It is not common to manually configure MAC addresses for an interface, but if you have special needs to do so, then this pool is used to assign a unique MAC address to each interface.

Examples

The following example adds a MAC address pool with 8 MAC addresses, and assigns it to the GigabitEthernet 0/0 interface:

```
ciscoasa(config)# mac-address pool pool1 000C.F142.4CD1 - 000C.F142.4CD7
ciscoasa(config)# interface gigabitethernet 0/0
ciscoasa(config-ifc)# mac-address cluster-pool pool1
```

Related Commands

Command	Description
interface	Configures an interface.
mac-address	Configures a MAC address for an interface.

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-tableaging-time*timeout_value*
no mac-address-table aging-time

Syntax Description

timeout_value	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.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.7(1)	You can now configure this command in routed mode when using Integrated Routing and Bridging.

Usage Guidelines

No usage guidelines.

Examples

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

```
ciscoasa(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.

Command	Description
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. 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 ASA drops the traffic and generates a system message.

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

Syntax Description

interface_name	The source bridge group member interface.
mac_address	The MAC address you want to add to the table.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.7(1)	You can now configure this command in routed mode when using Integrated Routing and Bridging.

Examples

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

```
ciscoasa(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 disable

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

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

Syntax Description

interface_name	The bridge group member interface on which you want to disable MAC learning.
disable	Disables MAC learning.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.7(1)	You can now configure this command in routed mode when using Integrated Routing and Bridging.

Examples

The following example disables MAC learning on the outside interface:

```
ciscoasa(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.
show mac-address-table	Shows the MAC address table, including dynamic and static entries.

Command	Description
show running-config mac-learn	Shows the **mac-learn** configuration.

mac-learn flood

To enable flooding for unknown MAC addresses for non IPv4/IPv6 packets, use the **mac-learn flood** command in global configuration mode. To disable MAC address flooding, use the **no** form of this command.

mac-learn flood
no mac-learn flood

Command Default

Flooding is disabled.

Command Modes

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

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

Command History

Release	Modification
9.7(1)	This command was added.

Examples

The following example enables MAC flooding:

```
ciscoasa(config)# mac-learn flood
```

Related Commands

Command	Description
clear configure mac-learn	Sets the **mac-learn** configuration to the default.
mac-address-table static	Adds static MAC address entries to the MAC address table.
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 to exempt MAC addresses from authentication and/or authorization, use the **mac-list** command in global configuration mode. To remove a MAC list entry, use the **no** form of this command.

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

Syntax Description

deny	Indicates that traffic matching this MAC address does not match the MAC list and is subject to both authentication and authorization when specified in the **aaa mac-exempt** command. You might need to add a deny entry to the MAC list if you permit a range of MAC addresses using a MAC address mask such as ffff.ffff.0000, and you want to force a MAC address in that range to be authenticated and authorized.
id	Specifies a hexadecimal MAC access list number. To group a set of MAC addresses, enter the **mac-list** command as many times as needed with the same ID value. The order of entries matters, because the packet uses the first entry it matches, as opposed to a best match scenario. If you have a permit entry, and you want to deny an address that is allowed by the permit entry, be sure to enter the deny entry before the permit entry.
mac	Specifies the source MAC address in 12-digit hexadecimal form; that is, nnnn.nnnn.nnnn
macmask	Specifies the portion of the MAC address that should be used for matching. For example, ffff.ffff.ffff matches the MAC address exactly. ffff.ffff.0000 matches only the first 8 digits.
permit	Indicates that traffic matching this MAC address matches the MAC list and is exempt from both authentication and authorization when specified in the **aaa mac-exempt** command.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

To enable MAC address exemption from authentication and authorization, use the **aaa mac-exempt** command. You can only add one instance of the **aaa mac-exempt** command, so be sure that your MAC list includes all the MAC addresses you want to exempt. You can create multiple MAC lists, but you can only use one at a time.

Examples

The following example bypasses authentication for a single MAC address:

```
ciscoasa(config)# mac-list abc permit 00a0.c95d.0282 ffff.ffff.ffff
ciscoasa(config)# aaa mac-exempt match abc
```

The following entry bypasses authentication for all Cisco IP Phones, which have the hardware ID 0003.E3:

```
ciscoasa(config)# mac-list acd permit 0003.E300.0000 FFFF.FF00.0000
ciscoasa(config)# aaa mac-exempt match acd
```

The following example bypasses authentication for a a group of MAC addresses except for 00a0.c95d.02b2. Enter the deny statement before the permit statement, because 00a0.c95d.02b2 matches the permit statement as well, and if it is first, the deny statement will never be matched.

```
ciscoasa(config)# mac-list 1 deny 00a0.c95d.0282 ffff.ffff.ffff
ciscoasa(config)# mac-list 1 permit 00a0.c95d.0000 ffff.ffff.0000
ciscoasa(config)# aaa mac-exempt match 1
```

Related Commands

Command	Description
aaa authentication	Enables user authentication.
aaa authorization	Enables user authorization services.
aaa mac-exempt	Exempts a list of MAC addresses from authentication and authorization.
clear configure mac-list	Removes a list of MAC addresses previously specified by the **mac-list** command.
show running-config mac-list	Displays a list of MAC addresses previously specified in the **mac-list** command.

mail-relay

To configure a local domain name, use the **mail-relay** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

mail-relay domain_name action { **drop-connection** | **log** }
no mail-relay domain_name action { **drop-connection** | **log** }

Syntax Description

domain_name	Specifies the domain name.
drop-connection	Closes the connection.
log	Generates a system log message.

Command Default

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
Parameters configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to configure a mail relay for a specific domain:

```
ciscoasa(config)# policy-map type inspect esmtp esmtp_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# mail-relay mail action drop-connection
```

Related Commands

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

management-access

To allow management access to an interface other than the one from which you entered the ASA when using VPN, use the **management-access** command in global configuration mode. To disable management 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 ASA from another interface. A physical or virtual interface can be specified.

Command Default

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	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.
9.9.(2)	Virtual interfaces can now be specified.
9.14(1)	SNMP is no longer supported.
9.17(1)	If you use the CiscoSSH stack (the **ssh stack ciscossh** command), then this feature is not supported for SSH.

Usage Guidelines

This command allows you to connect to an interface other than the one you entered the ASA from when using a full tunnel IPsec VPN or SSL VPN client (AnyConnect 2.x client, SVC 1.x) or across a site-to-site IPsec tunnel. You can use Telnet, SSH, Ping, or ASDM to connect to an ASA interface. You can also use a management access interface as the source interface for syslog messages sent through the VPN tunnel.

You can define only one management-access interface.

In 9.5(1) and later, due to routing considerations with the separate management and data routing tables, the VPN termination interface and the management access interface need to be the same type: both need to be management-only interfaces or regular data interfaces. Therefore, do not configure management-access on a management-only interface except in the rare instance that the VPN termination interface is management-only.

If you use the CiscoSSH stack (the **ssh stack ciscossh** command), then this feature is not supported for SSH.

This feature is not supported for SNMP in 9.14(1) and later. For SNMP over VPN, we recommend enabling SNMP on a loopback interface in 9.18(2) and later. You don't need the management-access feature enabled to use SNMP on the loopback interface. Loopback also works for SSH.

When using identity NAT between the management-access interface network and VPN networks (a common NAT configuration for VPN traffic), you must specify the **nat** command **route-lookup** keyword. Without route lookup, the ASA sends traffic out the interface specified in the **nat** command, regardless of what the routing table says. For example, you configure **management-access inside**, so a VPN user entering on the outside can manage the inside interface. If the identity **nat** command specifies **(inside,outside)**, then you do not want the ASA to send the management traffic out to the inside network; it will never return to the inside interface IP address. The route lookup option lets the ASA send the traffic directly to the inside interface IP address instead of to the inside network. For traffic from the VPN client to a host on the inside network, the route lookup option will still result in the correct egress interface (inside), so normal traffic flow is not affected.

Examples

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

```
ciscoasa(config)# management-access inside
```

Related Commands

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

management-only

To set an interface to accept management traffic only, use the **management-only** command in interface configuration mode. To allow through traffic, use the **no** form of this command.

management-only [**individual**]
no management-only [**individual**]

Syntax Description

individual	For the Firepower 9300 ASA security module cluster, you must specify the **individual** keyword for a management interface when in Spanned interface mode.

Command Default

The Management *n* /*n* interface, if available for your model, is set to management-only mode 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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	The placement of this command in the running configuration has been moved to the top of the interface section to support ASA clustering, which has special exemptions for management interfaces.
9.4(1.152)	The **individual** keyword was added.

Usage Guidelines

Most models include a dedicated management interface called Management *n* /*n* , which is meant to support traffic to the ASA. However, you can configure any interface to be a management-only interface using the **management-only** command.

Note For all models except the ASA 5585-X, you cannot disable management-only mode for the Management interface. By default, this command is always enabled.

In transparent firewall mode, in addition to the maximum allowed through-traffic interfaces, you can also use the Management interface (either the physical interface, a subinterface (if supported for your model), or an EtherChannel interface comprised of Management interfaces (if you have multiple Management interfaces)) as a separate management interface. You cannot use any other interface types as management interfaces.

If your model does not include a Management interface, you must manage the transparent firewall from a data interface.

In multiple context mode, you cannot share any interfaces, including the Management interface, across contexts. To provide management per context, you can create subinterfaces of the Management interface and allocate a Management subinterface to each context. Note that except for the ASA 5585-X, the management interface does not allow subinterfaces, so for per-context management, you must connect to a data interface.

The management interface is not part of a normal bridge group. Note that for operational purposes, it is part of a non-configurable bridge group.

Examples

The following example disables management-only mode on the Management interface:

```
ciscoasa(config)# interface management0/0
ciscoasa(config-if)# no management-only
```

The following example enables management-only mode on a subinterface:

```
ciscoasa(config)# interface gigabitethernet0/2.1
ciscoasa(config-subif)# management-only
```

Related Commands

Command	Description
interface	Configures an interface and enters interface configuration mode.

map-domain

To configure a Mapping Address and Port (MAP) domain, use the **map-domain** command in global configuration mode. Use the **no** form of this command to delete the MAP domain.

map-domain*name*
no map-domain *name*

Syntax Description

name	The name of the MAP domain, which is an alphanumeric string up to 48 characters. The name can also include the following special characters: period (.), slash (/), and colon (:).

Command Default

No defaults.

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 mode	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
9.13(1)	This command was introduced.

Usage Guidelines

Mapping Address and Port (MAP) is primarily a feature for use in service provider (SP) networks. The service provider can operate an IPv6-only network, the MAP domain, while supporting IPv4-only subscribers and their need to communicate with IPv4-only sites on the public Internet. MAP is defined in RFC7597, RFC7598, and RFC7599.

For the service provider, within the MAP domain, the benefit of MAP over NAT46 is that the substitution of an IPv6 address for the subscriber's IPv4 address (and back again to IPv4 at the SP network edge) is stateless. This provides greater efficiency within the SP network compared to NAT46.

There are two MAP techniques, MAP-Translation (MAP-T) and MAP-Encapsulation (MAP-E). The ASA supports MAP-T; MAP-E is not supported.

To configure MAP-T, you create one or more domains. When you configure MAP-T on customer edge (CE) and border relay (BR) devices, ensure that you use the same parameters for each device that will participate in each domain.

You can configure up to 25 MAP-T domains. In multiple-context mode, you can configure up to 25 domains per context.

Examples

The following example creates a MAP-T domain named 1 and configures the translation rules for the domain.

```
ciscoasa(config)# map-domain 1

ciscoasa(config-map-domain)# default-mapping-rule 2001:DB8:CAFE:CAFE::/64

ciscoasa(config-map-domain)# basic-mapping-rule

ciscoasa(config-map-domain-bmr)# ipv4-prefix 192.168.3.0 255.255.255.0

ciscoasa(config-map-domain-bmr)# ipv6-prefix 2001:cafe:cafe:1::/64

ciscoasa(config-map-domain-bmr)# start-port 1024

ciscoasa(config-map-domain-bmr)# share-ratio 16
```

Related Commands

Commands	Description
basic-mapping-rule	Configures the basic mapping rule for a MAP domain.
default-mapping-rule	Configures the default mapping rule for a MAP domain.
ipv4-prefix	Configures the IPv4 prefix for the basic mapping rule in a MAP domain.
ipv6-prefix	Configures the IPv6 prefix for the basic mapping rule in a MAP domain.
map-domain	Configures a Mapping Address and Port (MAP) domain.
share-ratio	Configures the number of ports in the basic mapping rule in a MAP domain.
show map-domain	Displays information about Mapping Address and Port (MAP) domains.
start-port	Configures the starting port for the basic mapping rule in a MAP domain.

map-name

To map a user-defined attribute name to a Cisco attribute name, use the **map-name** command in ldap-attribute-map configuration mode.

To remove this mapping, use the **no** form of this command.

map-name *user-attribute-name Cisco-attribute-name*
no map-name *user-attribute-name Cisco-attribute-name*

Syntax Description

user-attribute-name	Specifies the user-defined attribute name that you are mapping to the Cisco attribute.
Cisco-attribute-name	Specifies the Cisco attribute name that you are mapping to the user-defined name.

Command Default

By default, no name mappings exist.

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
ldap-attribute-map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.1(1)	This command was added.

Usage Guidelines

With the **map-name** command, you can map your own attribute names to Cisco attribute names. You can then bind the resulting attribute map to an LDAP server. Your typical steps would include:

1. Use the **ldap attribute-map** command in global configuration mode to create an unpopulated attribute map. This commands enters ldap-attribute-map configuration mode.

2. Use the **map-name** and **map-value** commands in ldap-attribute-map configuration mode to populate the attribute map.

3. Use the **ldap-attribute-map** command in aaa-server host mode to bind the attribute map to an LDAP server. Note the hyphen after “ldap” in this command.

Note To use the attribute mapping features correctly, you need to understand both the Cisco LDAP attribute names and values as well as the user-defined attribute names and values.

Examples

The following example commands map a user-defined attribute name Hours to the Cisco attribute name cVPN3000-Access-Hours in the LDAP attribute map myldapmap:

```
ciscoasa(config)# ldap attribute-map myldapmap
ciscoasa(config-ldap-attribute-map)# map-name Hours cVPN3000-Access-Hours
ciscoasa(config-ldap-attribute-map)#
```

Within ldap-attribute-map configuration mode, you can enter “?” to display the complete list of Cisco LDAP attribute names:

```
ciscoasa(config-ldap-attribute-map)# map-name <name>
ldap mode commands/options:
cisco-attribute-names:
  cVPN3000-Access-Hours
  cVPN3000-Allow-Network-Extension-Mode
  cVPN3000-Auth-Service-Type
  cVPN3000-Authenticated-User-Idle-Timeout
  cVPN3000-Authorization-Required
  cVPN3000-Authorization-Type
 :
 :
  cVPN3000-X509-Cert-Data
ciscoasa(config-ldap-attribute-map)#
```

Related Commands

Command	Description
ldap attribute-map (global configuration mode)	Creates and names an LDAP attribute map for mapping user-defined attribute names to Cisco LDAP attribute names.
ldap-attribute-map (aaa-server host mode)	Binds an LDAP attribute map to an LDAP server.
map-value	Maps a user-defined attribute value to a Cisco attribute.
show running-config ldap attribute-map	Displays a specific running LDAP attribute map or all running attribute maps.
clear configure ldap attribute-map	Removes all LDAP attribute maps.

mapping-service (Deprecated)

To configure a mapping service for the Cisco Intercompany Media Engine proxy, use the **mapping-service** command in UC-IME configuration mode. To remove the mapping service from the proxy, use the **no** form of this command.

mapping-service listening-interface *interface* [**listening-port** *port*] **uc-ime-interface** *interface*
no mapping-service listening-interface *interface* [**listening-port** *port*] **uc-ime-interface** *interface*

Syntax Description

interface	Specifies the name of the interface to be used for the listening interface or uc-ime interface.
listening-interface	Configures the interface on which the ASA listens for the mapping requests.
listening-port	(Optional) Configures the listening port for the mapping service.
port	(Optional) Specifies the TCP port number on which the ASA listens for the mapping requests. The port number must be 1024 or higher to avoid conflicts with other services on the device, such as Telnet or SSH. By default, the port number is TCP 8060.
uc-ime-interface	Configures the interface that connects to the remote Cisco UCM.

Command Default

By default the mapping-service for off-path deployments of the Cisco Intercompany Media Engine proxy listens on TCP port 8060.

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
UC-IME configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.3(1)	This command was added.
9.4(1)	This command was deprecated along with all **uc-ime** mode commands.

Usage Guidelines

For an off-path deployment of the Cisco Intercompany Media Engine proxy on the ASA, adds the mapping service to the proxy configuration. To configure the mapping service, you must specify the outside interface (remote enterprise side) on which to listen for mapping requests and the interface that connects to the remote Cisco UCM.

Note You can only configure one mapping server for the Cisco Intercompany Media Engine proxy.

You configure the mapping service when the Cisco Intercompany Media Engine proxy is configured for an off-path deployment.

In an off path deployment, inbound and outbound Cisco Intercompany Media Engine calls pass through an adaptive security appliance enabled with the Cisco Intercompany Media Engine proxy. The adaptive security appliance is located in the DMZ and configured to support primarily Cisco Intercompany Media Engine. Normal Internet-facing traffic does not flow through this ASA.

For all inbound calls, the signaling is directed to the ASA because destined Cisco UCMs are configured with the global IP address on the ASA. For outbound calls, the called party could be any IP address on the Internet; therefore, the ASA is configured with a mapping service that dynamically provides an internal IP address on the ASA for each global IP address of the called party on the Internet.

Cisco UCM sends all outbound calls directly to the mapped internal IP address on the adaptive security appliance instead of the global IP address of the called party on the Internet. The ASA then forwards the calls to the global IP address of the called party.

Examples

The following example shows ...:

```
ciscoasa
(config)# uc-ime offpath_uc-ime_proxy
ciscoasa(config-uc-ime)# media-termination ime-media-term
ciscoasa(config-uc-ime)# ucm address 192.168.10.30 trunk-security-mode non-secure
ciscoasa(config-uc-ime)# ticket epoch 1 password password1234
ciscoasa(config-uc-ime)# fallback monitoring timer 120
ciscoasa(config-uc-ime)# fallback hold-down timer 30
ciscoasa(config-uc-ime)# mapping-service listening-interface inside listening-port 8060
uc-ime-interface outside
```

Related Commands

Command	Description
show running-config uc-ime	Shows the running configuration of the Cisco Intercompany Media Engine proxy.
show uc-ime	Displays statistical or detailed information about fallback-notifications, mapping-service-sessions, and signaling-sessions.
uc-ime	Creates the Cisco Intercompany Media Engine proxy instance on the ASA.

map-value

To map a user-defined value to a Cisco LDAP value, use the **map-value** command in ldap-attribute-map configuration mode. To delete an entry within a map, use the **no** form of this command.

map-value *user-attribute-name user-value-string Cisco-value-string*
no map-value *user-attribute-name user-value-string Cisco-value-string*

Syntax Description

Cisco-value-string	Specifies the Cisco value string for the Cisco attribute.
user-attribute-name	Specifies the user-defined attribute name that you are mapping to the Cisco attribute name.
user-value-string	Specifies the user-defined value string that you are mapping to the Cisco attribute value.

Command Default

By default, there are no user-defined values mapped to Cisco attributes.

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
ldap-attribute-map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.1(1)	This command was added.

Usage Guidelines

With the **map-value** command, you can map your own attribute values to Cisco attribute names and values. You can then bind the resulting attribute map to an LDAP server. Your typical steps would include:

1. Use the **ldap attribute-map** command in global configuration mode to create an unpopulated attribute map. This commands enters ldap-attribute-map configuration mode.
2. Use the **map-name** and **map-value** commands in ldap-attribute-map configuration mode to populate the attribute map.
3. Use the **ldap-attribute-map** command in aaa-server host mode to bind the attribute map to an LDAP server. Note the hyphen after “ldap” in this command.

Note To use the attribute mapping features correctly, you need to understand both the Cisco LDAP attribute names and values as well as the user-defined attribute names and values.

Examples

The following example, entered in ldap-attribute-map configuration mode, sets the user-defined value of the user attribute Hours to a user-defined time policy named workDay and a Cisco-defined time policy named Daytime:

```
ciscoasa(config)# ldap attribute-map myldapmap
ciscoasa(config-ldap-attribute-map)# map-value Hours workDay Daytime
ciscoasa(config-ldap-attribute-map)#
```

Related Commands

Command	Description
ldap attribute-map (global configuration mode)	Creates and names an LDAP attribute map for mapping user-defined attribute names to Cisco LDAP attribute names.
ldap-attribute-map (aaa-server host mode)	Binds an LDAP attribute map to an LDAP server.
map-name	Maps a user-defined LDAP attribute name with a Cisco LDAP attribute name.
show running-config ldap attribute-map	Displays a specific running LDAP attribute map or all running attribute maps.
clear configure ldap attribute-map	Removes all LDAP maps.

mask

When using the Modular Policy Framework, mask out part of the packet that matches a **match** command or class map by using the **mask** command in match or class configuration mode. This mask action is available in an inspection policy map (the **policy-map type inspect** command) for application traffic; however, not all applications allow this action. For example, you can you use **mask** command for the DNS application inspection to mask a header flag before allowing the traffic through the ASA. To disable this action, use the no form of this command.

mask [**log**]
no mask [**log**]

Syntax Description

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

Command Default

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
Match and class configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

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

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

Examples

The following example masks the RD and RA flags in the DNS header before allowing the traffic through the ASA:

```
ciscoasa(config-cmap)# policy-map type inspect dns dns-map1
ciscoasa(config-pmap-c)# match header-flag RD
ciscoasa(config-pmap-c)# mask log
ciscoasa(config-pmap-c)# match header-flag RA
ciscoasa(config-pmap-c)# mask log
```

Related Commands

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

mask-banner

To obfuscate the server banner, use the **mask-banner** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

mask-banner
no mask-banner

Syntax Description

This command has no arguments or keywords.

Command Default

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
Parameters configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to mask the server banner:

```
ciscoasa(config)# policy-map type inspect esmtp esmtp_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# mask-banner
```

Related Commands

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

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.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

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 ASA to replace the FTP server replies to the syst command with Xs:

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

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.

Commands	Description
request-command deny	Specifies FTP commands to disallow.

match access-list

When using the Modular Policy Framework, use an access list to identify traffic to which you want to apply actions by using the **match access-list** command in class-map configuration mode. To remove the **match access-list** command, use the **no** form of this command.

match access-list *access_list_name*
no match access-list *access_list_name*

Syntax Description

access_list_name	Specifies the name of an access list to be used as match criteria.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

Configuring Modular Policy Framework consists of four tasks:

1. Identify the Layer 3 and 4 traffic to which you want to apply actions using the **class-map** command.

After you enter the **class-map** command, you can enter the **match access-list** command to identify the traffic. Alternatively, you can enter a different type of **match** command, such as the **match port** command. You can only include one **match access-list** command in the class map, and you cannot combine it with other types of **match** commands. The exception is if you define the **matchdefault-inspection-traffic** command which matches the default TCP and UDP ports used by all applications that the ASA can inspect, then you can narrow the traffic to match using a **match access-list** command. Because the **match default-inspection-traffic** command specifies the ports to match, any ports in the access list are ignored.

1. (Application inspection only) Define special actions for application inspection traffic using the **policy-map type inspect** command.
2. Apply actions to the Layer 3 and 4 traffic using the **policy-map** command.
3. Activate the actions on an interface using the **service-policy** command.

Examples

The following example creates three Layer 3/4 class maps that match three access lists:

```
ciscoasa(config)# access-list udp permit udp any any
ciscoasa(config)# access-list tcp permit tcp any any
ciscoasa(config)# access-list host_foo permit ip any 10.1.1.1 255.255.255.255
ciscoasa(config)# class-map all_udp
ciscoasa(config-cmap)# description "This class-map matches all UDP traffic"
ciscoasa(config-cmap)# match access-list udp
ciscoasa(config-cmap)# class-map all_tcp
ciscoasa(config-cmap)# description "This class-map matches all TCP traffic"
ciscoasa(config-cmap)# match access-list tcp
ciscoasa(config-cmap)# class-map to_server
ciscoasa(config-cmap)# description "This class-map matches all traffic to server 10.1.1.1"
ciscoasa(config-cmap)# match access-list host_foo
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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

When using the Modular Policy Framework, match all traffic to which you want to apply actions by using the **match any** command in class-map configuration mode. To remove the **match any** command, use the **no** form of this command.

match any
no match any

Syntax Description

This command has no arguments or keywords.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

Configuring Modular Policy Framework consists of four tasks:

1. Identify the Layer 3 and 4 traffic to which you want to apply actions using the **class-map** command.

After you enter the **class-map** command, you can enter the **match any** command to identify all traffic. Alternatively, you can enter a different type of **match** command, such as the **match port** command. You cannot combine the **match any** command with other types of **match** commands.

1. (Application inspection only) Define special actions for application inspection traffic using the **policy-map type inspect** command.
2. Apply actions to the Layer 3 and 4 traffic using the **policy-map** command.
3. Activate the actions on an interface using the **service-policy** command.

Examples

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

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

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
match access-list	Matches traffic according to an access list.
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 apn

To configure a match condition for an access point name in GTP messages, use the **match apn** command in policy-map configuration mode. To remove the match condition, use the **no** form of this command.

match [**not**] **apn regex** { *regex_name* | **class** *regex_class_name* }
no match [**not**] **apn regex** [*regex_name* | **class** *regex_class_name*]

Syntax Description

regex_name	Specifies a regular expression.
class *regex_class_name*	Specifies a regular expression class map.

Command Default

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
Policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

This command can be configured in a GTP policy map.

Examples

The following example shows how to configure a match condition for an access point name in an GTP inspection policy map:

```
ciscoasa(config-pmap)# match apn class gtp_regex_apn
```

Related Commands

Command	Description
inspect gtp	Configures inspection of GTP traffic.

match application-id

To configure a match condition for the Diameter application identifier of Diameter messages, use the **match application-id** command in class-map or policy-map configuration mode. To remove the match condition, use the **no** form of this command.

match [**not**] **application-id** *app_id* [*app_id_2*]
no match [**not**] **application-id** *app_id* [*app_id_2*]

Syntax Description

app_id	The Diameter application name or number (0-4294967295). If there is a range of consecutively-numbered applications that you want to match, you can include a second ID. You can define the range by application name or number, and it applies to all the numbers between the first and second IDs.

Command Default

Diameter inspection allows all applications.

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 or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.5(2)	This command was added.

Usage Guidelines

This command can be configured in a Diameter inspection class map or policy map. Use it to filter traffic based on Diameter application ID. You can then drop the packet, drop the connection, or log matching traffic.

These applications are registered with the IANA. Following are the core supported applications, but you can filter on other applications. Use the CLI help for a list of application names.

- **3gpp-rx-ts29214** (16777236)
- **3gpp-s6a** (16777251)
- **3gpp-s9** (16777267)
- **common-message** (0). This is the base Diameter protocol.

The IETF has a list of registered applications, command codes, and attribute-value pairs at http://www.iana.org/assignments/aaa-parameters/aaa-parameters.xhtml , although Diameter inspection does not support all listed items. See the 3GPP web site for their technical specifications.

Examples

The following example shows how to configure a match condition for the 3gpp-s6a and 3gpp-s13 application IDs.

```
ciscoasa(config)# class-map type inspect diameter match-any log_app

ciscoasa(config-cmap)# match application-id 3gpp-s6a

ciscoasa(config-cmap)# match application-id 3gpp-s13
```

Related Commands

Command	Description
class-map type inspect	Creates an inspection class map.
inspect diameter	Enables Diameter inspection.
policy-map type inspect	Creates an inspection policy map.

match as-path

To match a BGP autonomous system path access list, use the match as-path command in route-map configuration mode. To remove a path list entry, use the no form of this command.

match as-path *path-list-number*
no match as-path *path-list-number*

Syntax Description

path-list-number	Autonomous system path access list number.

Command Default

No path lists 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	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
9.2(1)	This command was added.

Usage Guidelines

The values set by the match as-path and set weight commands override global values. For example, the weights assigned with the match as-path and set weight route-map configuration commands override the weight assigned using the neighbor weight command.

A route map can have several parts. Any route that does not match at least one match clause relating to a route-map command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route-map section with an explicit match specified. It can accept more than one path-list-name.

Examples

The following example sets the autonomous system path to match BGP autonomous system path access list as-path-acl:

```
ciscoasa(config)# route-map IGP2BGP
ciscoasa(config-route-map)# match as-path 23
```

Related Commands

Command	Description
set-weight	Specifies the BGP weight for the routing table.
neighbor-weight	Assigns a weight to a neighbor connection.

match avp

To configure a match condition for a Diameter attribute-value pair (AVP) in Diameter messages, use the **match avp** command in class-map or policy-map configuration mode. To remove the match condition, use the **no** form of this command.

To match AVP by attribute only:

match [**not**] **avp** *code* [*code-2*] [**vendor-id** *id_number*]
no match [**not**] **avp** *code* [*code-2*] [**vendor-id** *id_number*]

To match an AVP based on the value of the attribute:

match [**not**] **avp** *code* [**vendor-id** *id_number*] *value*
no match [**not**] **avp** *code* [**vendor-id** *id_number*] *value*

Syntax Description

code	The name or number (1-4294967295) of an attribute-value pair. For the first code, you can specify the name of a custom AVP or one that is registered in RFCs or 3GPP technical specifications and is directly supported in the software. If you want to match a range of AVP, specify the second code by number only. If you want to match an AVP by its value, you cannot specify a second code. See the CLI help for a list of AVP names.
value	The value portion of the AVP. You can configure this only if the data type of the AVP is supported. For example, you can specify an IP address for AVP that have the address data type. For detailed information on how to configure this parameter, see the Usage section below.
vendor-id *id_number*	(Optional.) The ID number of the vendor to also match, from 0-4294967295. For example, the 3GPP vendor ID is 10415, the IETF is 0.

Command Default

Diameter inspection allows all AVP.

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 or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.5(2)	This command was added.

Usage Guidelines

This command can be configured in a Diameter inspection class map or policy map. Use it to filter traffic based on Diameter AVP. You can then drop the packet, drop the connection, or log matching traffic.

Use the CLI help for a list of AVP names. The IETF has a list of registered applications, command codes, and attribute-value pairs at https://www.iana.org/assignments/aaa-parameters/aaa-parameters.xhtml , although Diameter inspection does not support all listed items. See the 3GPP web site for their technical specifications.

If you are configuring a value match, following are the specific syntax of the value option for the supported data types:

- Diameter Identity, Diameter URI, Octet String—Use regular expression or regular expression class objects to match these data types.

{**regex** *regex_name* | **class** *regex_class*}

- Address—Specify the IPv4 or IPv6 address to match. For example, 10.100.10.10 or 2001:DB8::0DB8:800:200C:417A.
- Time—Specify the start and end dates and time. Both are required. Time is in 24-hour format.

date *year month day* **time** *hh*:*mm*:*ss* **date** *year month day* **time** *hh*:*mm*:*ss*

For example:

date 2015 feb 5 time 12:00:00 date 2015 mar 9 time 12:00:00

- Numeric—Specify a range of numbers:

range *number_1 number_2*

The valid number range depends on the data type:

- Integer32: -2147483647 to 2147483647
- Integer64: -9223372036854775807 to 9223372036854775807
- Unsigned32: 0 to 4294967295
- Unsigned64: 0 to 18446744073709551615
- Float32: decimal point representation with 8 digit precision
- Float64: decimal point representation with 16 digit precision

Examples

The following example shows how to configure a match condition for a specific IP address that appears on the host-ip-address AVP on Capability Exchange Request/Answer command messages.

```
ciscoasa(config)# class-map type inspect diameter match-all block-ip

ciscoasa(config-cmap)# match command-code cer-cea

ciscoasa(config-cmap)# match avp host-ip-address 1.1.1.1
```

Related Commands

Command	Description
class-map type inspect	Creates an inspection class map.

Command	Description
diameter	Creates custom attribute-value pairs.
inspect diameter	Enables Diameter inspection.
policy-map type inspect	Creates an inspection policy map.

match body

To configure a match condition on the length or length of a line of an ESMTP body message, use the **match body** command in class-map or policy-map configuration mode. To remove a configured section, use the **no** form of this command.

match [**not**] **body** [**length** | **line length**] **gt** *bytes*
no match [**not**] **body** [**length** | **line length**] **gt** *bytes*

Syntax Description

length	Specifies the length of an ESMTP body message.
line length	Specifies the length of a line of an ESMTP body message.
bytes	Specifies the number to match in bytes.

Command Default

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 or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to configure a match condition for a body line length in an ESMTP inspection policy map:

```
ciscoasa
(config)#
 policy-map type inspect esmtp esmtp_map
```

ciscoasa (config-pmap)# **match body line length gt 1000**

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.

Command	Description
clear configure class-map	Removes all class maps.
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 called-party

To configure a match condition on the H.323 called party, use the **match called-party** command in policy-map configuration mode. To disable this feature, use the **no** form of this command.

match [**not**] **called-party** [**regex** *regex*]
no match [**not**] **match** [**not**] **called-party** [**regex** *regex*]

Syntax Description

regex regex	Specifies to match on the regular expression.

Command Default

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
Policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to configure a match condition for the called party in an H.323 inspection class map:

```
ciscoasa(config-cmap)# match called-party regex caller1
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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 calling-party

To configure a match condition on the H.323 calling party, use the **match calling-party** command in policy-map configuration mode. To disable this feature, use the **no** form of this command.

match [**not**] **calling-party** [**regex** *regex*]
no match [**not**] **match** [**not**] **calling-party** [**regex** *regex*]

Syntax Description

regex regex	Specifies to match on the regular expression.

Command Default

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
Policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to configure a match condition for the calling party in an H.323 inspection class map:

```
ciscoasa(config-cmap)# match calling-party regex caller1
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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 certificate

To configure a certificate match rule, use the **match certificate** command in crypto ca trustpoint configuration mode. To remove the rule from the configuration, use the **no** form of this command.

match certificate *map-name* [**override ocsp** [**trustpoint** *trustpoint-name*] *seq-num* **url** *URL* | **override cdp** *seq-num* **url** *URL*]
no match certificate *map-name* [**override ocsp** [*seq-num* **url** *URL*] | **override cdp** [*seq-num* **url** *URL*]]

Syntax Description

map-name	Specifies the name of the certificate map to match to this rule. You must configure the certificate map before configuring a match rule. The maximum length is 65 characters.
override ocsp	Specifies that the purpose of the rule is to override an OCSP URL in a certificate.
seq-num	Sets the priority for this match rule. The valid range is from 1 to 10000. The ASA evaluates the match rule with the lowest sequence number first, followed by higher numbers until it finds a match.
trustpoint	(Optional) Specifies using a trustpoint for verifying the OCSP responder certificate.
trustpoint-name	(Optional) Identifies the trustpoint to use with the override to validate responder certificates.
url	Specifies accessing a URL for OCSP revocation status.
URL	Identifies the URL to access for OCSP revocation status.
override cdp	Specifies that the purpose of the rule is to override a CRL URL in a certificate.
seq-num	Sets the rank of each URL in the list. Specifies a value from 1 to 5. The ASA tries the URL at lowest rank (1) first.
url	Specifies accessing a URL for CRL revocation status.
URL	The URL to access the CRL revocation status.

Command Default

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
crypto ca trustpoint configuration	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
7.2(1)	This command was added.
9.13(1)	Provision to configure cdp override was added.
9.15(1)	Prior to this release, static CDPs can be mapped uniquely to each certificate in a chain that is being validated. However, only one such mapping was supported for each certificate. From this release, the match certificate cdp override command accepts multiple instances for the same map name.

Usage Guidelines

During the PKI certificate validation process, the ASA checks certificate revocation status to maintain security by using either CRL checking or Online Certificate Status Protocol (OCSP). With CRL checking, the ASA retrieves, parses, and caches CRLs, which provide a complete list of revoked certificates. OCSP offers a more scalable method of checking revocation status because OCSP localizes certificate status on a validation authority, which it queries for the status of a specific certificate.

Certificate match rules let you configure OCSP URL overrides, which specify a URL to check for revocation status, rather than the URL in the AIA field of the remote user certificate. Match rules also let you configure trustpoints to use to validate OCSP responder certificates, which let the ASA validate responder certificates from any CA, including self-signed certificates and certificates external to the validation path of the client certificate.

Similar to OCSP, you can use the **match certificate** command to configure CDP URL overrides. This command supports the identification of static CDP URLs through the certificate map. For each certificate that needs CRL validation, CRLs are retrieved based on the CDP extension in the certificate and any URLs that are mapped in this configuration. The **policy** command in the **config-ca-crl** submode can be used to exclude the CDPs from the certificate or the static CDPs.

You can now configure multiple static CDPs to a single map. To remove individual instances, in the **no** form of the command, specify the URL and sequence numbers. Ensure that the specified URL and sequence numbers are the same values that you had configured. If you are not mentioning any specific information, all the entries for the map will be removed. The provision to have or to remove multiple instances for a map is not applicable for OCSP.

When configuring OCSP, be aware of the following requirements:

- You can configure multiple match rules within a trustpoint configuration, but you can have only one match rule for each crypto ca certificate map. You can, however, configure multiple crypto ca certificate maps and associate them with the same trustpoint.
- You must configure the certificate map before configuring a match rule.
- To configure a trustpoint to validate a self-signed OCSP responder certificates, you import the self-signed responder certificate into its own trustpoint as a trusted CA certificate. Then you configure the **match certificate** command in the client certificate validating trustpoint to use the trustpoint that contains the self-signed OCSP responder certificate to validate the responder certificate. The same applies for validating responder certificates external to the validation path of the client certificate.
- A trustpoint can validate both the client certificate and the responder certificate if the same CA issues both of them. But if different CAs issue the client and responder certificates, you need to configure two trustpoints, one trustpoint for each certificate.
- The OCSP server (responder) certificate typically signs the OCSP response. After receiving the response, the ASA tries to verify the responder certificate. The CA normally sets the lifetime of its OCSP responder

certificate to a relatively short period to minimize the chance of it being compromised.The CA typically also includes an ocsp-no-check extension in the responder certificate indicating that this certificate does not need revocation status checking. But if this extension is not present, the ASA tries to check its revocation status using the same method specified in the trustpoint. If the responder certificate is not verifiable, revocation checks fails. To avoid this possibility, use the **revocation-check none** command when configuring the responder certificate validating trustpoint, and use the **revocation-check ocsp** command when configuring the client certificate.

- If the ASA does not find a match, it uses the URL specified in the **ocsp url** command. If you have not configured the **ocsp url** command, the ASA uses the AIA field of the remote user certificate. If the certificate does not have an AIA extension, revocation status checking fails.

Examples

The following example shows how to create a certificate match rule for a trustpoint called newtrust. The rule has a map name called mymap, a sequence number of 4, a trustpoint called mytrust, and specifies a URL of 10.22.184.22.

```
ciscoasa(config)# crypto ca trustpoint
 newtrust
ciscoasa(config-ca-trustpoint)# match certificate mymap override ocsp trustpoint mytrust 4
 url 10.22.184.22
ciscoasa(config-ca-trustpoint)#
```

The following example shows how to configure a crypto ca certificate map, and then a match certificate rule to identify a trustpoint that contains a CA certificate to validate the responder certificate. This certificate is necessary if the CA identified in the newtrust trustpoint does not issue an OCSP responder certificate.

1. Configure the certificate map that identifies the client certificates to which the map rule applies. In this example, the name of the certificate map is mymap and the sequence number is 1. Any client certificate with a subject-name that contains a CN attribute equal to mycert matches the mymap entry.

```
ciscoasa(config)# crypto ca certificate map mymap 1 subject-name attr cn eq mycert
ciscoasa(config-ca-cert-map)# subject-name attr cn eq mycert
ciscoasa(config-ca-cert-map)#
```

2. Configure a trustpoint that contains the CA certificate to use to validate the OCSP responder certificate. In the case of self-signed certificates, this is the self-signed certificate itself, which is imported and locally trusted. You can also obtain a certificate for this purpose through external CA enrollment. When prompted to do so, paste in the CA certificate.

```
ciscoasa(config-ca-cert-map)# exit
ciscoasa(config)# crypto ca trustpoint mytrust
ciscoasa(config-ca-trustpoint)# enroll terminal
ciscoasa(config-ca-trustpoint)# crypto ca authenticate mytrust
Enter the base 64 encoded CA certificate.
End with the word “quit” on a line by itself
```

```
MIIBnjCCAQcCBEPOpG4wDQYJKoZIhvcNAQEEBQAwFzEVMBMGA1UEAxQMNjMuNjcu
NzIuMTg4MB4XDTA2MDExODIwMjYyMloXDTA5MDExNzIwMjYyMlowFzEVMBMGA1UE
AxQMNjMuNjcuNzIuMTg4MIGdMA0GCSqGSIb3DQEBAQUAA4GLADCBhwKBgQDnXUHv
7//x1xEAOYfUzJmH5sr/NuxAbA5gTUbYA3pcE0KZHt761N+/8xGxC3DIVB8u7T/b
v8RqzqpmZYguveV9cLQK5tsxqW3DysMU/4/qUGPfkVZ0iKPCgpIAWmq2ojhCFPyx
ywsDsjl6YamF8mpMoruvwOuaUOsAK6KO54vy0QIBAzANBgkqhkiG9w0BAQQFAAOB
gQCSOihb2NH6mga2eLqEsFP1oVbBteSkEAm+NRCDK7ud1l3D6UC01EgtkJ81QtCk
tvX2T2Y/5sdNW4gfueavbyqYDbk4yxCKaofPp1ffAD9rrUFQJM1uQX14wclPCcAN
e7kR+rscOKYBSgVHrseqdB8+6QW5NF7f2dd+tSMvHtUMNw==
quit
```

```
INFO: Certificate has the following attributes:
Fingerprint: 7100d897 05914652 25b2f0fc e773df42
Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.

% Certificate successfully imported
```

3. Configure the original trustpoint, newtrust, with OCSP as the revocation checking method. Then set a match rule that includes the certificate map, mymap, and the self-signed trustpoint, mytrust, configured in Step 2.

```
ciscoasa(config)# crypto ca trustpoint newtrust
ciscoasa(config-ca-trustpoint)# enroll terminal
ciscoasa(config-ca-trustpoint)# crypto ca authenticate newtrust

Enter the base 64 encoded CA certificate.
End with the word “quit” on a line by itself
ywsDsjl6YamF8mpMoruvwOuaUOsAK6KO54vy0QIBAzANBgkqhkiG9w0BAQQFAAOB
gQCSOihb2NH6mga2eLqEsFP1oVbBteSkEAm+NRCDK7ud1l3D6UC01EgtkJ81QtCk
AxQMNjMuNjcuNzIuMTg4MIGdMA0GCSqGSIb3DQEBAQUAA4GLADCBhwKBgQDnXUHv
7//x1xEAOYfUzJmH5sr/NuxAbA5gTUbYA3pcE0KZHt761N+/8xGxC3DIVB8u7T/b
gQCSOihb2NH6mga2eLqEsFP1oVbBteSkEAm+NRCDK7ud1l3D6UC01EgtkJ81QtCk
tvX2T2Y/5sdNW4gfueavbyqYDbk4yxCKaofPp1ffAD9rrUFQJM1uQX14wclPCcAN
NzIuMTg4MB4XDTA2MDExODIwMjYyMloXDTA5MDExNzIwMjYyMlowFzEVMBMGA1UE
OPIBnjCCAQcCBEPOpG4wDQYJKoZIhvcNAQEEBQAwFzEVMBMGA1UEAxQMNjMuNjcu
e7kR+rscOKYBSgVHrseqdB8+6QW5NF7f2dd+tSMvHtUMNw==
quit
INFO: Certificate has the following attributes:
Fingerprint: 9508g897 82914638 435f9f0fc x9y2p42
Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.

% Certificate successfully imported
ciscoasa(config)# crypto ca trustpoint newtrust
ciscoasa(config-ca-trustpoint)# revocation-check ocsp
ciscoasa(config-ca-trustpoint)# match certificate mymap override ocsp trustpoint mytrust
 4 url 10.22.184.22
```

Any connection that uses the newtrust trustpoint for client certificate authentication checks to see if the client certificate matches the attribute rules specified in the mymap certificate map. If so, the ASA accesses the OCSP responder at 10.22.184.22 for certificate revocation status, then uses the mytrust trustpoint to validate the responder certificate.

Note The newtrust trustpoint is configured to perform revocation checking via OCSP for the client certificates. However, the mytrust trustpoint is configured for the default revocation-check method, which is none. As a result, no revocation checking is performed on the OCSP responder certificate.

The following example shows configuring a match certificate rule using CDP. The rule has a map name called test, with 1, 2, and 3 as sequence numbers, and static URLs. While selecting CDPs for a certificate, ASA selects the 3 CDPs for any certificate that matches the certificate map named *test*. If the ASA determines that a CRL is needed while validating the certificate, the URLs are tried in the given sequence until a CRL is successfully retrieved.

```
ciscoasa(config)# crypto ca trustpoint newtrust
ciscoasa(config-ca-trustpoint)# match certificate test override cdp 1 url http://1.1.1.1
ciscoasa(config-ca-trustpoint)# match certificate test override cdp 2 url http://1.1.1.2
ciscoasa(config-ca-trustpoint)# match certificate test override cdp 3 url http://1.1.1.3
ciscoasa(config-ca-trustpoint)#
```

Related Commands

Command	Description
crypto ca certificate map	Creates crypto ca certificate maps. Use this command in global configuration mode.
crypto ca trustpoint	Enters crypto ca trustpoint configuration mode. Use this command in global configuration mode.
ocsp disable-nonce	Disables the nonce extension of the OCSP request.
ocsp url	Specifies the OCSP server to use to check all certificates associated with a trustpoint.
revocation-check	Specifies the method(s) to use for revocation checking and the order in which to try them.

match certificate allow expired-certificate (deprecated)

To allow an administrator to exempt certain certificates from expiration checking, use the **match certificate allow expired-certificate** command in ca-trustpool configuration mode. To disable the exemption of certain certificates, use the **no** form of this command.

match certificate < **map** > **allow expired-certificate**
no match certificate < **map** > **allow expired-certificate**

Syntax Description

allow	Allows expired certificate to be accepted.

Command Default

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
Ca-trustpool configuration	• Yes	• Yes	• Yes	—	—

Command History

Release	Modification
9.0(1)	This command was added.
9.13(1)	This command was removed.

Usage Guidelines

The trustpool match commands leverage the certificate map objects to configure certificate specific exceptions or overrides to the global trustpool policy. The match rules are written relative to the certificate that is being validated.

Related Commands

Command	Description
match certificate skip revocation check	Exempts certain certificates from revocation checking.

match certificate skip revocation-check

To allow an administrator to exempt certain certificates from revocation checking, use the **match certificate skip revocation-check** command in ca-trustpool configuration mode. To disable the exemption from revocation checking, use the **no** form of this command.

matchcertificatemapskiprevocation-check
nomatchcertificatemapskiprevocation-check

Syntax Description

This command has no arguments or keywords.

Command Default

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
Ca-trustpool configuration	• Yes	• Yes	• Yes	—	—

Command History

Release	Modification
9.0(1)	This command was added.

Usage Guidelines

The trustpool match commands leverage the certificate map objects to configure certificate specific exceptions or overrides to the global trustpool policy. The match rules are written relative to the certificate that is being validated.

Examples

The following example shows skipping the validity check for the certificate with the Subject DN common name of “mycompany123.”

```
crypto ca certificate map mycompany 1subject-name attr cn eq mycompany123
crypto ca trustpool policymatch certificate mycompany skip revocation-check
```

Related Commands

Command	Description
match certificate allow expired-certificate	Exempts certain certificates from expiration checking.

match cmd

To configure a match condition on the ESMTP command verb, use the **match cmd** command in policy-map configuration mode. To disable this feature, use the **no** form of this command.

match [**not**] **cmd** [**verb** *verb* | **line length gt** *bytes* | **RCPT count gt** *recipients_number*]
no match [**not**] **cmd** [**verb** *verb* | **line length gt** *bytes* | **RCPT count gt** *recipients_number*]

Syntax Description

verb verb	Specifies the ESMTP command verb.
line length gt bytes	Specifies the length of a line.
RCPT count gt recipients_number	Specifies the number of recipient email addresses.

Command Default

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
Policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to configure a match condition in an ESMTP inspection policy map for the verb (method) NOOP exchanged in the ESMTP transaction:

```
ciscoasa(config-pmap)# match cmd verb NOOP
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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 command-code

To configure a match condition for the Diameter command code of Diameter messages, use the **match command-code** command in class-map or policy-map configuration mode. To remove the match condition, use the **no** form of this command.

match [**not**] **command-code** *code* [*code_2*]
no match [**not**] **command-code** *code* [*code_2*]

Syntax Description

code	The Diameter command code name or number (0-4294967295). If there is a range of consecutively-numbered command codes that you want to match, you can include a second code. You can define the range by command code name or number, and it applies to all the numbers between the first and second codes. See the CLI help for a list of command code names.

Command Default

Diameter inspection allows all command codes.

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 or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.5(2)	This command was added.

Usage Guidelines

This command can be configured in a Diameter inspection class map or policy map. Use it to filter traffic based on Diameter command code. You can then drop the packet, drop the connection, or log matching traffic.

The IETF has a list of registered applications, command codes, and attribute-value pairs athttp://www.iana.org/assignments/aaa-parameters/aaa-parameters.xhtml , although Diameter inspection does not support all listed items. See the 3GPP web site for their technical specifications.

Examples

The following example shows how to configure a match condition for a specific IP address that appears on the host-ip-address AVP on Capability Exchange Request/Answer command messages.

```
ciscoasa(config)# class-map type inspect diameter match-all block-ip

ciscoasa(config-cmap)# match command-code cer-cea

ciscoasa(config-cmap)# match avp host-ip-address 1.1.1.1
```

Related Commands

Command	Description
class-map type inspect	Creates an inspection class map.
inspect diameter	Enables Diameter inspection.
policy-map type inspect	Creates an inspection policy map.

match community

To match a Border Gateway Protocol (BGP) community, use the match community command in route-map configuration mode. To remove the match community command from the configuration file and restore the system to its default condition where the software removes the BGP community list entry, use the no form of this command.

match community { *standard-list-number* | *expanded-list-number* | *community-list-name* [**exact**] }
no match community { *standard-list-number* | *expanded-list-number* | *community-list-name* [**exact**] }

Syntax Description

standard-list-number	Specifies a standard community list number from 1 to 99 that identifies one or more permit or deny groups of communities.
expanded-list-number	Specifies an expanded community list number from 100 to 500 that identifies one or more permit or deny groups of communities
community-list-name	The community list name.
exact	(Optional) Indicates that an exact match is required. All of the communities and only those communities specified must be present.

Command Default

No community list is matched by the route map.

Command Modes

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

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

Command History

Release	Modification
9.2(1)	This command was added.

Usage Guidelines

A route map can have several parts. Any route that does not match at least one match command relating to a route-map command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route-map section with an explicit match specified.

Matching based on community list number is one of the types of match commands applicable to BGP.

Examples

The following example shows that the routes matching community list 1 will have the weight set to 100. Any route that has community 109 will have the weight set to 100.

```
ciscoasa(config)#  community-list 1 permit 109
ciscoasa(config)# route-map set_weight
ciscoasa(config-route-map)#  match community 1
ciscoasa(config-route-map)# set weight 100
```

The following example shows that the routes matching community list 1 will have the weight set to 200. Any route that has community 109 alone will have the weight set to 200.

```
ciscoasa(config)#  community-list 1 permit 109
ciscoasa(config)# route-map set_weight
ciscoasa(config-route-map)# match community 1 exact
ciscoasa(config-route-map)# set weight 200
```

In the following example, the routes that match community list LIST_NAME will have the weight set to 100. Any route that has community 101 alone will have the weight set to 100.

```
ciscoasa(config)#  community-list LIST_NAME permit 101
ciscoasa(config)# route-map set_weight
ciscoasa(config-route-map)# match community LIST_NAME
ciscoasa(config-route-map)# set weight 100
```

The following example shows that the routes that match expanded community list 500. Any route that has extended community 1 will have the weight set to 150.

```
ciscoasa(config)#  community-list 500 permit [0-9]*
ciscoasa(config)# route-map MAP_NAME permit 10
ciscoasa(config-route-map)# match extcommunity 500
ciscoasa(config-route-map)# set weight 150
```

Related Commands

Command	Description
set-weight	Specifies the BGP weight for the routing table.
community-list	Creates or configures a BGP community list.

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.

matchdefault-inspection-traffic
nomatchdefault-inspection-traffic

Syntax Description

This command has no arguments or keywords.

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.6(2)	TCP/53 was added for DNS over TCP inspection, which is not enabled by default. Default ports for M3UA and STUN were also added.

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:

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

Default traffic for inspections are as follows:

Inspection Type	Protocol Type	Source Port	Destination Port
ctiqbe	tcp	N/A	2748
dcerpc	tcp	N/A	135
diameter	tcp, sctp	N/A	3868
dns	udp, tcp	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
im	tcp	N/A	1-65539
ip-options	rsvp	N/A	N/A
ipsec-pass-thru	udp	N/A	500
m3ua	sctp	N/A	2905
mgcp	udp	2427,2727	2427,2727
netbios	udp	137-138	N/A
radius-accounting	udp	N/A	1646
rpc	udp	111	111
rsh	tcp	N/A	514
rtsp	tcp	N/A	554
sctp	sctp	any	any
sip	tcp, udp	N/A	5060
skinny	tcp	N/A	2000

smtp	tcp	N/A	25
sqlnet	tcp	N/A	1521
stun	tcp, udp	N/A	3478
tftp	udp	N/A	69
waas	tcp	N/A	1-65535
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:

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

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 dns-class

To configure a match condition for the Domain System Class in a DNS Resource Record or Question section, use the **match dns-class** command in class-map or policy-map configuration mode. To remove a configured class, use the **no** form of this command.

match [**not**] **dns-class** { **eq** *c_well_known* | *c_val* } { **range** *c_val1 c_val2* }
no match [**not**] **dns-class** { **eq** *c_well_known* | *c_val* } { **range** *c_val1 c_val2* }

Syntax Description

eq	Specifies an exact match.
c_well_known	Specifies DNS class by well-known name, IN.
c_val	Specifies an arbitrary value in the DNS class field (0-65535).
range	Specifies a range.
c_val1 c_val2	Specifies values in a range match. Each value between 0 and 65535.

Command Default

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
Class-map or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

By default, this command inspects all fields (questions and RRs) of a DNS message and matches the specified class. Both DNS query and response are examined.

The match can be narrowed down to the question portion of a DNS query by the following two commands: match not header-flag QR and match question.

This command can be configured within a DNS class map or policy map. Only one entry can be entered within a DNS class-map.

Examples

The following example shows how to configure a match condition for a DNS class in a DNS inspection policy map:

```
ciscoasa(config)# policy-map type inspect dns preset_dns_map
ciscoasa(config-pmap)# match dns-class eq IN
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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 dns-type

To configure a match condition for a DNS type, including Query type and RR type, use the **match dns-type** command in class-map or policy-map configuration mode. To remove a configured dns type, use the **no** form of this command.

match [**not**] **dns-type** { **eq** *t_well_known* | *t_val* } { **range** *t_val1 t_val2* }
no match [**not**] **dns-type** { **eq** *t_well_known* | *t_val* } { **range** *t_val1 t_val2* }

Syntax Description

eq	Specifies an exact match.
t_well_known	Specifies DNS type by well-known name: A, NS, CNAME, SOA, TSIG, IXFR, or AXFR.
t_val	Specifies an arbitrary value in the DNS type field (0-65535).
range	Specifies a range.
t_val1 t_val2	Specifies values in a range match. Each value between 0 and 65535.

Command Default

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
Class-map or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

By default, this command inspects all sections of a DNS message (questions and RRs) and matches the specified type. Both DNS query and response are examined.

The match can be narrowed down to the question portion of a DNS query by the following two commands: match not header-flag QR and match question.

This command can be configured within a DNS class map or policy map. Only one entry can be entered within a DNS class-map.

Examples

The following example shows how to configure a match condition for a DNS type in a DNS inspection policy map:

```
ciscoasa(config)# policy-map type inspect dns preset_dns_map
ciscoasa(config-pmap)# match dns-type eq a
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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 domain-name

To configure a match condition for a DNS message domain name list, use the **match domain-name** command in class-map or policy-map configuration mode. To remove a configured section, use the **no** form of this command.

match [**not**] **domain-name regex** *regex_id*
match [**not**] **domain-name regex class** *class_id*
no match [**not**] **domain-name regex** *regex_id*
no match [**not**] **domain-name regex class** *class_id*

Syntax Description

regex	Specifies a regular expression.
regex_id	Specifies the regular expression ID.
class	Specifies the class map that contains multiple regular expression entries.
class_id	Specifies the regular expression class map ID.

Command Default

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
Class-map or policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

This command matches domain names in the DNS message against predefined list. Compressed domain names will be expanded before matching. The match condition can be narrowed down to a particular field in conjunction with other DNS match commands.

This command can be configured within a DNS class map or policy map. Only one entry can be entered within a DNS class-map.

Examples

The following example shows how to match the DNS domain name in a DNS inspection policy map:

```
ciscoasa(config)# policy-map type inspect dns preset_dns_map
ciscoasa(config-pmap)# match domain-name regex
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
clear configure class-map	Removes all class maps.
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 dpc

To configure a match condition for the destination point code (DPC) of M3UA data messages, use the **match dpc** command in policy-map configuration mode. To remove the match condition, use the **no** form of this command.

match [**not**] **dpc** *code*
no match [**not**] **dpc** *code*

Syntax Description

code	The destination point code in *zone -region -sp* format.

Command Default

M3UA inspection allows all destination point codes.

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
policy map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.6(2)	This command was added.

Usage Guidelines

You can configure this command in an M3UA inspection policy map. You can drop packets based on the destination point code. Point code is in *zone -region -sp* format, where the possible values for each element depend on the SS7 variant. You define the variant on the **ss7 variant** command in the policy map.

- ITU—Point codes are 14 bit in 3-8-3 format. The value ranges are [0-7]-[0-255]-[0-7]. This is the default SS7 variant.
- ANSI—Point codes are 24 bit in 8-8-8 format. The value ranges are [0-255]-[0-255]-[0-255].
- Japan—Point codes are 16 bit in 5-4-7 format. The value ranges are [0-31]-[0-15]-[0-127].
- China—Point codes are 24 bit in 8-8-8 format. The value ranges are [0-255]-[0-255]-[0-255].

Examples

The following example shows how to configure a match condition for a specific destination point code for ITU.

```
ciscoasa(config)# policy-map type inspect m3ua m3ua-map
ciscoasa(config-pmap)# match dpc 1-5-1
ciscoasa(config-pmap-c)# drop log
ciscoasa(config-pmap-c)# parameters
ciscoasa(config-pmap-p)# ss7 variant ITU
```

Related Commands

Command	Description
inspect m3ua	Enables M3UA inspection.
match opc	Matches the M3UA originating point code.
policy-map type inspect	Creates an inspection policy map.
ss7 variant	Identifies the SS7 variant to use in the policy map.

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

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

Command Default

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	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

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:

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

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.