ca - cld

cache

To enter cache mode and set values for caching attributes, enter the **cache** command in webvpn configuration mode. To remove all cache related commands from the configuration and reset them to their default values, enter the **no** form of this command.

cache
no cache

Command Default

Disabled.

Command Modes

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

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

Command History

Release	Modification
7.1(1)	This command was added.
9.5(2)	The default changed from enabled to disabled.

Usage Guidelines

Caching stores frequently reused objects in the system cache, which reduces the need to perform repeated rewriting and compressing of content. It reduces traffic between WebVPN and both the remote servers and end-user browsers, so that many applications run much more efficiently.

Note Enabling the content cache may cause some systems to become less reliable. If you experience random crashes after enabling the content cache, disable it.

The following example shows how to enter cache mode:

```
ciscoasa
(config)#
 webvpn
ciscoasa
(config-webvpn)#
 cache
hostname(config-webvpn-cache)#
```

Related Commands

Command	Description
cache-static-content	Caches content not subject to rewriting.
disable	Disables caching.
expiry-time	Configures the expiration time for caching objects without revalidating them.
lmfactor	Sets a revalidation policy for caching objects that have only the last-modified timestamp.
max-object-size	Defines the maximum size of an object to cache.
min-object-size	Defines the minimum size of an object to cache.

ca-check

To configure the basic constraints extension and set the CA flag in a trustpoint certificate, use the **ca-check** command in crypto ca trustpoint configuration mode. To not set the basic constraints extension and CA flag, use the **no** form of this command.

ca-check
no ca-check

Syntax Description

This command has no arguments or keywords.

Command Default

By default, the basic constraints extension and CA flag are set. You must use the **no** form to disable them.

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	—

Command History

Release	Modification
9.4(1)	This command was added.

Usage Guidelines

The basic constraints extension identifies whether the subject of the certificate is a Certificate Authority (CA), in which case the certificate can be used to sign other certificates. The CA flag is part of this extension. The presence of these items in a certificate indicates that the certificate's public key can be used to validate certificate signatures.

Examples

The following example shows how to disable the CA flag and basic constraints extension.

```
ciscoasa(config)# crypto ca trustpoint newtrust

ciscoasa(config-ca-trustpoint)# no ca-check

ciscoasa(config-ca-trustpoint)#
```

Related Commands

Command	Description
crypto ca trustpoint	Enters crypto ca trustpoint configuration mode. Use this command in global configuration mode.

cache-static-content

To cache all static content used for Clientless SSL VPN connections, enter the cache-static-content command in webvpn cache configuration mode. To disable caching of static content, enter the **no** form of this command.

cache-static-content enable
no cache-static-content enable

Syntax Description

enable	Enables the loading of all static content into cache memory.

Command Default

Disabled.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was added.

Usage Guidelines

Configuring the security appliance to store all cache-able static content in the appliance cache increases the performance of back-end SSL VPN connections. Static content includes objects not rewritten by the security appliance, such as PDF files and images.

Examples

The following example enables caching of static content:

```
ciscoasa(config-webvpn-cache)# cache-static-content enable
```

Related Commands

Command	Description
disable	Disables caching.
expiry-time	Configures the expiration time for caching objects without revalidating them.

cache-time

To specify in minutes how long to allow a CRL to remain in the cache before considering it stale, use the **cache-time** command in ca-crl configuration mode, which is accessible from crypto ca trustpoint configuration mode. To return to the default value, use the **no** form of this command.

cache-time *refresh-time*
no cache-time

Syntax Description

refresh-time	Specifies the number of minutes to allow a CRL to remain in the cache. The range is 1 - 1440 minutes. If the NextUpdate field is not present in the CRL, the CRL is not cached.

Command Default

The default setting is 60 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
Ca-crl configuration	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
7.0(1)	This command was added.

Examples

The following example enters ca-crl configuration mode, and specifies a cache time refresh value of 10 minutes for trustpoint central:

```
ciscoasa(configure)# crypto ca trustpoint central
ciscoasa(ca-trustpoint)# crl configure
ciscoasa(ca-crl)# cache-time 10
ciscoasa(ca-crl)#
```

Related Commands

Command	Description
crl configure	Enters crl configuration mode.
crypto ca trustpoint	Enters trustpoint configuration mode.
enforcenextupdate	Specifies how to handle the NextUpdate CRL field in a certificate.

call-agent

To specify a group of call agents, use the **call-agent** command in mgcp map configuration mode. To remove the configuration, use the **no** form of this command.

call-agent *ip_address group_id*
no call-agent *ip_address group_id*

Syntax Description

group_id	The ID of the call agent group, from 0 to 2147483647.
ip_address	The IP address of the gateway.

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
Mgcp map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

Use the call-agent command to specify a group of call agents that can manage one or more gateways. The call agent group information is used to open connections for call agents in the group (other than the one to which a gateway sends a command) so that any of the call agents can send the response. Call agents with the same >*group_id* belong to the same group. A call agent may belong to more than one group.

Examples

The following example allows call agents 10.10.11.5 and 10.10.11.6 to control gateway 10.10.10.115, and allows call agents 10.10.11.7 and 10.10.11.8 to control both gateways 10.10.10.116 and 10.10.10.117:

```
ciscoasa(config)# mgcp-map mgcp_inbound
ciscoasa(config-mgcp-map)# call-agent 10.10.11.5 101
ciscoasa(config-mgcp-map)# call-agent 10.10.11.6 101
ciscoasa(config-mgcp-map)# call-agent 10.10.11.7 102
ciscoasa(config-mgcp-map)# call-agent 10.10.11.8 102
ciscoasa(config-mgcp-map)# gateway 10.10.10.115 101
ciscoasa(config-mgcp-map)# gateway 10.10.10.116 102
ciscoasa(config-mgcp-map)# gateway 10.10.10.117 102
```

Related Commands

Commands	Description
debug mgcp	Enables the display of debugging information for MGCP.
mgcp-map	Defines an MGCP map and enables MGCP map configuration mode.
show mgcp	Displays MGCP configuration and session information.

call-duration-limit

To configure the call duration for an H.323 call, use the **call-duration-limit** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

call-duration-limit hh:mm:ss
no call-duration-limit hh:mm:ss

Syntax Description

hh:mm:ss	Specifies the duration in hours, minutes, and seconds.

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 the call duration for an H.323 call:

```
ciscoasa(config)# policy-map type inspect h323 h323_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# call-duration-limit 0:1:0
```

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 or 4 policy map.
show running-config policy-map	Displays all current policy map configurations.

call-party-numbers

To enforce sending call party numbers during an H.323 call setup, use the **call-party-numbers** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

call-party-numbers
no call-party-numbers

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 enforce call party numbers during call setup for an H.323 call:

```
ciscoasa(config)# policy-map type inspect h323 h323_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# call-party-numbers
```

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 or 4 policy map.
show running-config policy-map	Displays all current policy map configurations.

call-home

To enter call home configuration mode, use the **call-home** command in global configuration mode.

call-home

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
Global configuration	• Yes	• Yes	• Yes	—	• Yes

Command History

Release	Modification
8.2(2)	This command was added.

Usage Guidelines

After you enter the **call-home** command, the prompt changes to hostname (cfg-call-home)#, and you have access to the following Call Home configuration commands:

- [no] alert-group {group name | all}—Enables or disables the Smart Call Home group. The default is enabled for all alert-groups.group name: Syslog, diagnostic, environment, inventory, configuration, snapshot, threat, telemetry, test.
- [no] contact-e-mail-addr e-mail-address—Specifies the customer contact e-mail address. This field is required.e-mail-address: A customer e-mail address of up to 127 characters.
- [no] contact-name contact name—Specifies the customer name.e-mail-address: A customer name of up to 127 characters.
- [no] contract-id contract-id-string—Specifies customer contract identification.contract-id-string: An identification number up to 128 characters. Spaces are allowed, but you must use quotes around the string if it includes spaces.
- copy profile src-profile-name dest-profile-name—Copies the content of an existing profile (**src-profile-name**) to a new profile (**dest-profile-name**).src-profile-name: An existing profile name of up to 23 characters.dest-profile-name: A new profile name of up to 23 characters.
- rename profile src-profile-name dest-profile-name—Changes the name of an existing profile.src-profile-name: An existing profile name of up to 23 characters.dest-profile-name: A new profile name of up to 23 characters.
- source-interface—Specifies the source interface.

- no configuration all—Clears the Smart Call-home configuration.[no] customer-id customer-id-string—Specifies the customer ID.customer-id-string: A customer ID of up to 64 characters. This field is required for XML format messages.
- [no] event-queue-size queue_size—Specifies the event queue size.queue-size: The number of events from 5-60. The default is 10.
- [no] mail-server ip-address | name priority 1-100 all—Specifies the SMTP mail server. Customers can specify up to five mail servers. At least one mail server is required for using e-mail transport for Smart Call Home messages. ip-address: The IPv4 or IPv6 address of the mail server.name: The hostname of the mail server.1-100: The priority of the mail server. The lower the number, the higher the priority.
- [no] phone-number phone-number-string—Specifies the customer phone number. This field is optional.phone-number-string: The phone number.
- [no] rate-limit msg-count—Specifies the number of messages that Smart Call Home can send per minute.msg-count: The number of messages per minute. The default is 10.
- [no] sender {from e-mail-address | reply-to e-mail-address} —Specifies the from/reply-to e-mail address of an e-mail message. This field is optional.e-mail-address: The from and reply-to e-mail address.
- [no] site-id site-id-string—Specifies the customer site ID. This field is optional.site-id-string: A site ID to identify the location of the customer.
- [no] street-address street-address—Specifies the customer address. This field is optional.street-address: A free-format string of up to 255 characters.
- [no] alert-group-config environment—Enters environment group configuration mode.[no] threshold {cpu | memory} low-high—Specifies the environmental resource threshold.low, high: Valid values are 0-100. The default is 85-90.
- [no] alert-group-config snapshot—Enters snapshot group configuration mode.system, user: To run the CLI in sysem or user context (available only in multimode).
- [no] add-command "cli command" [{system | user}] —Specifies CLI commands to capture in the snapshot group.cli command: The CLI command to be entered.system, user: To run the CLI in the system or in user context (available only in multiple mode). If both the system and user are not specified, the CLI will be run in both the system and user contexts. The default is the user context.
- All bullets below moved to profile command.
- [no] profile profile-name | no profile all—Creates, deletes, or edits a profile. Enters profile configuration mode and changes the prompt to hostname (cfg-call-home-profile)#. profile-name: A profile name of up to 20 characters.
- [no] active—Enables or disables a profile. The default is enabled.no destination address {e-mail | http} all | [no] destination {address {e-mail | http} e-mail-address | http-url [msg-format short-text | long-text | xml] | message-size-limit max-size | preferred-msg-format short-text | long-text | xml | transport-method e-mail | http}—Configures the destination, message size, message format, and transport method for the Smart Call Home message receiver. The default message format is XML, and the default enabled transport method is e-mail.e-mail-address: The e-mail address of the Smart Call Home receiver, which can be up to 100 characters.http-url: The HTTP or HTTPS URL.max-size: The maximum message size in bytes. 0 means no limit. The default is 5 MB.
- [no] subscribe-to-alert-group alert-group-name [severity {catastrophic | disaster | emergencies | alert | critical | errors | warning | notifications | informational | debugging}]—Subscribes to events of a group

with a specified severity level.alert-group-name: Syslog, diagnostic, environment, or threat are valid values.

- [no] subscribe-to-alert-group syslog [{severity {catastrophic | disaster | emergencies | alert | critical | errors | warning | notifications | informational | debugging} | message start [-end]}]—Subscribes to syslogs with a severity level or message ID.start-[end]: One syslog message ID or a range of syslog message IDs.

Note Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debugging** only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Also, use it during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood of increasing the processing overhead that will affect system use.

- [no] subscribe-to-alert-group inventory [periodic {daily | monthly day_of_month | weekly day_of_week [hh:mm]]—Subscribes to inventory events.day_of_month: Day of the month, 1-31.day_of_week: Day of the week (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday).hh, mm: Hours and minutes of a day, in 24-hour time.
- [no] subscribe-to-alert-group configuration [export full | minimum] [periodic {daily | monthly day_of_month | weekly day_of_week [hh:mm]]—Subscribes to configuration events.full: Configuration to export the running configuration, startup configuration, feature list, number of elements in an access list, and the context name in multimode.minimum: Configuration to export-only feature list, number of elements in an access list, and the context name in multimode.day_of_month: Day of the month, 1-31.day_of_week: Day of the week (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday).hh, mm: Hours and minutes of a day, in 24-hour time.
- [no] subscribe-to-alert-group telemetry periodic {hourly | daily | monthly day_of_month | weekly day_of_week [hh:mm]—Subscribes to telemetry periodic events.day_of_month: Day of the month, 1-31.day_of_week: Day of the week (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday).hh, mm: Hours and minutes of a day, in 24-hour time.
- [no] subscribe-to-alert-group snapshot periodic {interval minutes | hourly [mm] | daily | monthly day_of_month |weekly day_of_week [hh:mm]}—Subscribes to snapshot periodic events.minutes: The interval in minutes.day_of_month: Day of the month, 1-31.day_of_week: Day of the week (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday).hh, mm: Hours and minutes of a day, in 24-hour time.

Examples

The following example show how to configure contact information:

```
hostname(config)# call-home
hostname(cfg-call-home)# contact-e-mail-addr username@example.com
hostname(cfg-call-home)# customer-id Customer1234
hostname(cfg-call-home)# phone-number +1-800-555-0199
hostname(cfg-call-home)# site-id Site1
hostname(cfg-call-home)# street-address "1234 Any Street, Any city, Any state, 12345"
```

The following example shows how to configure the Call Home message rate-limit threshold:

```
hostname(config)# call-home
hostname(cfg-call-home)# rate-limit 50
```

The following example shows how to set the Call Home message rate-limit threshold to the default setting:

```
hostname(config)# call-home
hostname(cfg-call-home)# default
rate-limit
```

The following example shows how to create a new destination profile with the same configuration settings as an existing profile:

```
hostname(config)# call-home
hostname(cfg-call-home)# copy profile profile1 profile1a
```

The following example shows how to configure the general e-mail parameters, including a primary and secondary e-mail server:

```
hostname(config)# call-home
hostname(cfg-call-home)# mail-server smtp.example.com priority 1
hostname(cfg-call-home)# mail-server 192.168.0.1 priority 2
hostname(cfg-call-home)# sender from username@example.com
hostname(cfg-call-home)# sender reply-to username@example.com
```

Related Commands

Command	Description
alert-group	Enables an alert group.
profile	Enters call-home profile configuration mode.
show call-home	Displays Call Home configuration information.

call-home send

To execute a CLI command and e-mail the command output to a specified address, use the **call-home send** command in privileged EXEC mode.

call-home send cli command [**email** *email*] [**service-number** *service number*]

Syntax Description

cli-command	Specifies the CLI command to be executed. The command output is sent by e-mail.
email *email*	Specifies the e-mail address to which the CLI command output is sent. If no e-mail address is specified, the command output is sent to the Cisco TAC at attach@cisco.com.
service-number *service number*	Specifies an active TAC case number to which the command output pertains. This number is required only if no e-mail address (or a TAC e-mail address) is specified, and will appear in the e-mail subject line.

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
Privileged EXEC	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
8.2(2)	This command was added.

Usage Guidelines

This command causes the specified CLI command to be executed on the system. The specified CLI command must be enclosed in quotes (“”), and can be any **run** or **show** command, including commands for all modules.

The command output is then sent by e-mail to the specified e-mail address. If no e-mail address is specified, the command output is sent to the Cisco TAC at attach@cisco.com. The e-mail is sent in long text format with the service number, if specified, in the subject line.

Examples

The following example shows how to send a CLI command and have the command output e-mailed:

```
hostname# call-home send "show diagnostic result module all" email support@example.com
```

Related Commands

call-home	Enters call home configuration mode.

call-home test	Sends a Call Home test message that you define.
service call-home	Enables or disables Call Home.
show call-home	Displays call-home configuration information.

call-home send alert-group

To send a specific alert group message, use the **call-home send alert-group** command in privileged EXEC mode.

call-home send alert-group { **configuration** | **telemetry** | **inventory** | **group snapshot** } [**profile** *profile-name*]

Syntax Description

configuration	Sends the configuration alert-group message to the destination profile.
group snapshot	Sends the snapshot group.
inventory	Sends the inventory call-home message.
profile *profile-name*	(Optional) Specifies the name of the destination profile.
telemetry	Sends the diagnostic alert-group message to the destination profile for a specific module, slot/subslot, or slot/bay number.

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
Privileged EXEC	• Yes	• Yes	• Yes	—	• Yes

Command History

Release	Modification
8.2(2)	This command was added.

Usage Guidelines

If you do not specify the profile *profile-name*, the message is sent to all subscribed destination profiles.

Only the configuration, diagnostic, and inventory alert groups can be manually sent. The destination profile need not be subscribed to the alert group.

Examples

The following example shows how to send the configuration alert-group message to the destination profile:

```
hostname# call-home send alert-group configuration
```

The following example shows how to send the diagnostic alert-group message to the destination profile for a specific module, slot/subslot, or slot/bay number:

```
hostname# call-home send alert-group diagnostic module 3 5/2
```

The following example shows how to send the diagnostic alert-group message to all destination profiles for a specific module, slot/subslot, or slot/bay number:

```
hostname# call-home send alert-group diagnostic module 3 5/2 profile Ciscotac1
```

This example shows how to send the inventory call-home message:

```
hostname# call-home send alert-group inventory
```

Related Commands

call-home	Enters call home configuration mode.
call-home test	Sends a Call Home test message that you define.
service call-home	Enables or disables Call Home.
show call-home	Displays call-home configuration information.

call-home test

To manually send a Call Home test message using the configuration of a profile, use the **call-home test** command in privileged EXEC mode.

call-home test [*"test-message"*] **profile** *profile-name*

Syntax Description

profile *profile-name*	Specifies the name of the destination profile.
“*test-message*”	(Optional) Test message text.

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
Privileged EXEC	• Yes	• Yes	• Yes	—	• Yes

Command History

Release	Modification
8.2(2)	This command was added.

Usage Guidelines

This command sends a test message to the specified destination profile. If you enter test message text, you must enclose the text in quotes (“”) if it contains spaces. If you do not enter a message, a default message is sent.

Note The **call-home test** command is applicable only for Smart Software Managers that manage the device licenses online, and not for Smart Software Manager On-Prem servers.

Examples

The following example shows how to manually send a Call Home test message:

```
hostname# call-home test “test of the day” profile Ciscotac1
```

Related Commands

call-home	Enters call home configuration mode.
call-home send alert-group	Sends a specific alert group message.

service call-home	Enables or disables Call Home.
show call-home	Displays Call Home configuration information.

capability lls

The LLS capability is enabled by default. To explicitly enable the use of the Link-Local Signalling (LLS) data block in originated OSPF packets and re-enable OSPF NSF awareness, use the capability lls command in the router-configuration mode. To disable LLS and OSPF NSF awareness, use the no form of this command.

capability lls
no capability lls

Syntax Description

This command has no arguments or keywords.

Command Default

LLS capability 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
Router-configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.3(1)	This command was introduced.

Usage Guidelines

You might want to disable NSF awareness by disabling the use of the LLS data block in originated OSPF packets. You might want to disable NSF awareness if the router has no applications using LLS.

If NSF is configured and you try to disable LLS, you will receive the error message, “OSPF Non-Stop Forwarding (NSF) must be disabled first.”

If LLS is disabled and you try to configure NSF, you will receive the error message, “OSPF Link-Local Signaling (LLS) capability must be enabled first.”

Examples

The following example enables LLS support and OSPF awareness:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-router)# capability lls
```

Related Commands

capability opaque	Enable MPLS TE information to be flooded to through the network using opaque LSAs

capability opaque

To enable Multiprotocol Label Switching traffic engineering (MPLS TE) topology information to flood the network through opaque LSAs, use the capability opaque command in the router-configuration mode. To disable MPLS TE topology information flooding through opaque LSAs to the network, use the no form of the command.

capability opaque
no capability opaque

Syntax Description

This command has no arguments or keywords.

Command Default

Opaque LSAs are 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
Router-configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.3(1)	This command was introduced.

Usage Guidelines

The capability opaque command floods MPLS TE information (Types 1 and 4) through opaque LSAs of all scope (Types 9, 10, and 11).

Control opaque LSA support capability must be enabled for OSPF to support MPLS TE.

The MPLS TE topology information is flooded to the area through opaque LSAs by default.

Examples

The following example enables opaque capability:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-router)# capability opaque
```

Related Commands

capability lls	Enables use of LLS data-block in OSPF originated packets and enables OSPF NSF awareness.

captive-portal

To enable captive portal for the ASA FirePOWER module, use the **captive-portal** command in global configuration mode. To disable captive portal, use the **no** form of this command.

captive-portal { **global** | **interface** *name* } [**port** *number*]
no captive-portal { **global** | **interface** *name* } [**port** *number*]

Syntax Description

global	Enables captive portal globally on all interfaces.
interface *name*	Enables captive portal on the specified interface only. You can enter the command multiple times to enable it on more than one interface. You can use this approach if you are redirecting traffic for only a subset of interfaces to the ASA FirePOWER module.
port *number*	(Optional.) Sets the authentication proxy port to 1025 higher. Do not specify the keyword if you want to configure the default port, which is 885.

Command Default

The default port is 885 (TCP).

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
9.5(2)	This command was added.

Usage Guidelines

Captive portal works in conjunction with the identity policy defined on the ASA FirePOWER module.

For HTTP/HTTPS connections, you can define identity rules that collect user identification through active authentication. If you want to implement active authentication identity rules, you must configure captive portal on the ASA to act as the authentication proxy port. When a connection matches an identity rule that requests active authentication, the ASA FirePOWER module redirects the authentication request to the ASA interface IP address/captive portal. The default port is 885, which you can change.

If you do not enable captive portal for the authentication proxy, only passive authentication is available.

Examples

The following example enables captive portal globally on the default port 885:

```
ciscoasa(config)# captive-portal global
ciscoasa(config)#
```

Related Commands

Command	Description
sfr	Redirects traffic to the ASA FirePOWER module.
show running-config captive-portal	Displays the captive portal configuration.
show service-policy	Shows service policy statistics.

capture

To enable packet capture capabilities for packet sniffing and network fault isolation, use the **capture** command in privileged EXEC mode. To disable packet capture capabilities, use the **no** form of this command.

Capture network traffic:

capture *capture_name* [**type** { **asp-drop** [**all** | *drop-code*] | **tls-proxy** | **raw-data** | **isakmp** [**ikev1** | **ikev2**] | **inline-tag** [*tag*] | **webvpn user** *webvpn-user* }] [**access-list** *access_list_name* { **interface** { *interface_name* | **asa_dataplane asa_mgmt_plane** | **cplane** } } [**buffer** *buf_size*] [**ethernet-type** *type*] [**reeinject-hide**] [**packet-length** *bytes*] [**circular-buffer**] [**trace** [**trace-count** *number*]] [**real-time** [**dump**] [**detail**]] [**match** *protocol* { **host** *source-ip* | *source-ip mask* | **any** | **any4** | **any6** } [*operator src_port*] { **host** *dest_ip* | *dest_ip mask* | | **any** | **any4** | **any6** } [*operator dest_port*]] [**switch**] [**offload**] [**ivlan** *number*] [**ovlan** *number*]

Capture cluster control-link traffic:

capture *capture_name* { **type lacp interface** *interface_id* [**buffer** *buf_size*] [**packet-length** *bytes*] [**circular-buffer**] [**real-time** [**dump**] [**detail**]]

capture *capture_name* **interface cluster** [**buffer** *buf_size*] [**ethernet-type** *type*] [**packet-length** *bytes*] [**circular-buffer**] [**cp-cluster**] [**trace** [**trace-count** *number*]] [**real-time** [**dump**] [**detail**]] [**match** *protocol* { **host** *source-ip* | *source-ip mask* | **any** | **any4** | **any6** } [*operator src_port*] { **host** *dest_ip* | *dest_ip mask* | | **any** | **any4** | **any6** } [*operator dest_port*]]

Ingress switch capture packets for Secure Firewall 3100 model devices:

capture *capture_name* **switch interface** *interface_name* [**drop** { **disable** | **mac-filter** }]

Switch capture packets for Secure Firewall 4200 model devices:

capture *capture_name* **switch interface** *interface_name* [**direction** { { **both** | **egress** } [**drop disable**] | **ingress** [**drop** { **disable** | **mac-filter** }] }]

Note For Secure Firewall 4200 model devices, the **mac-filter** option is supported only for the ingress direction.

Capture packets cluster-wide:

cluster exec capture *capture_name* [**persist**] [**include-decrypted**]

Clear persistent packet traces cluster-wide:

cluster exec clear packet-trace

Remove the packet capture:

no capture *capture_name* [*arguments*]

Manually stop or start the packet capture:

capture *capture_name* **stop**

no capture *capture_name* **stop**

Syntax Description

access-list *access_list_name*	(Optional) Captures traffic that matches an access list. In multiple context mode, this is only available within a context.
any	Specifies all IPv4 traffic.
any4	Specifies all IPv4 traffic.
any6	Specifies all IPv6 traffic.
all	Captures all packets dropped by the accelerated security path.
asa_dataplane	Captures packets on the ASA backplane that pass between the ASA and a module that uses the backplane, such as the ASA FirePOWER module.
asp-drop *drop-code*	(Optional) Captures packets dropped by the accelerated security path. The *drop-code* specifies the type of traffic that is dropped by the accelerated security path. See the **show asp drop frame** command for a list of drop codes. You can enter this keyword with the **packet-length**, **circular-buffer**, and **buffer** keywords, but not with the **interface** or **ethernet-type** keyword. In a cluster, dropped forwarded data packets from one unit to another are also captured. In multiple context mode, when this option is issued in the system execution space, all dropped data packets are captured; when this option is issued in a context, only dropped data packets that enter from interfaces belonging to the context are captured.
buffer *buf_size*	(Optional) Defines the buffer size used to store the packet in bytes. Once the byte buffer is full, packet capture stops. When used in a cluster, this is the per-unit size, not the sum of all units.
capture_name	Specifies the name of the packet capture. Use the same name on multiple **capture** statements to capture multiple types of traffic. When you view the capture configuration using the **show capture** command, all options are combined on one line.
circular-buffer	(Optional) Overwrites the buffer, starting from the beginning, when the buffer is full.
cp-cluster	(Optional) Capture control packets on cluster interface.
direction	(Optional. Supported only on Secure Firewall 4200 model devices.) Specifies the direction of the switch traffic to be captured. It can be one of the following: • **both**—To capture switch bi-directional traffic • **egress**—To capture switch egressing traffic • **ingress**—To capture switch ingressing traffic

drop	Specifies the packet capture configuration of the mac-filter drop: • **disable**—To disable capture of packets dropped from switch. • **mac-filter**—To capture switch mac-filter drop. **Note** • For Secure Firewall 3100 model devices, drop is available when you select the interface. • For Secure Firewall 4200 model devices, the drop keyword is available only when you select the direction. However, the mac-filter option is supported only for the ingress packet capture direction.
ethernet-type *type*	(Optional) Selects an Ethernet type to capture. Supported Ethernet types include 8021Q, ARP, IP, IP6, LACP, PPPOED, PPPOES, RARP, and VLAN. An exception occurs with the 802.1Q or VLAN type. The 802.1Q tag is automatically skipped and the inner Ethernet type is used for matching.
host *ip*	Specifies the single IP address of the host to which the packet is being sent.
include-decrypted	(Optional) Captures decrypted IPsec packets which contain both normal and decrypted traffic once they enter the firewall device. It also captures packets of SSL decrypted traffic. However, the capture does not include the decrypted packets from VTI because they are available only on the VTI interface and not on the outside interface.
inline-tag *tag*	Specifies a tag for a particular SGT value or leaves it unspecified to capture a tagged packet with any SGT value.
interface *interface_name*	Sets the name of the interface on which to use packet capture. You must configure an interface for any packets to be captured except for **type asp-drop**. You can configure multiple interfaces using multiple **capture** commands with the same name. To capture packets on the dataplane, management plane, or control plane of an ASA, you can use the **interface** keyword with **asa_dataplane**, **asa_mgmt_plane**, or **cplane** as the interface name.You can specify **cluster** as the interface name to capture the traffic on the cluster control link interface. If the type **lacp** capture is configured, the interface name is the physical name.
ikev1 or **ikev2**	Captures only IKEv1 or IKEv2 protocol information.
isakmp	(Optional) Captures ISAKMP traffic for VPN connections. The ISAKMP subsystem does not have access to the upper layer protocols. The capture is a pseudo capture, with the physical, IP, and UDP layers combined together to satisfy a PCAP parser. The peer addresses are obtained from the SA exchange and are stored in the IP layer.
lacp	(Optional) Captures LACP traffic. If configured, the interface name is the physical interface name.
mask	The subnet mask for the IP address. When you specify a network mask, the method is different from the Cisco IOS software access-list command. The ASA uses a network mask (for example, 255.255.255.0 for a Class C mask). The Cisco IOS mask uses wildcard bits (for example, 0.0.0.255).

match *protocol*	Specifies the packets that match the five-tuple to allow filtering of those packets to be captured. You can use this keyword up to three times on one line.
operator	(Optional) Matches the port numbers used by the source or destination. The permitted operators are as follows: • lt—less than • gt—greater than • eq—equal to • **neq**—not equal to • **range**—range
packet-length *bytes*	(Optional) Sets the maximum number of bytes of each packet to store in the capture buffer.
persit	(Optional) Captures persistent packets on cluter units.
port	(Optional) If you set the protocol to tcp or udp, specifies the integer or name of a TCP or UDP port.
raw-data	(Optional) Captures inbound and outbound packets on one or more interfaces.
real-time	Displays the captured packets continuously in real-time. To terminate real-time packet capture, enter **Ctrl + c.** To permanently remove the capture, use the **no** form of this command. This option applies only to **raw-data**, **switch**, and **asp-drop** captures. This option is not supported when you use the **cluster exec capture** command.
reinject-hide	(Optional) Specifies that no reinjected packets will be captured. Applies only in a clustering environment.
stop	(Optional) Manually stops the capture without removing it. Use the **no** form of this command to start the capture.
tls-proxy	(Optional) Captures decrypted inbound and outbound data from TLS proxy on one or more interfaces.
trace *trace_count*	(Optional) Captures packet trace information, and the number of packets to capture. This option is used with an access list to insert trace packets into the data path to determine whether or not the packet has been processed as expected.
type	(Optional) Specifies the type of data captured.
user *webvpn-user*	(Optional) Specifies a username for a WebVPN capture.
webvpn	(Optional) Captures WebVPN data for a specific WebVPN connection.

Command Default

The defaults are as follows:

- The default **type** is **raw-data**
- The default buffer *size* is 512 KB .

- The default Ethernet type is IP packets.
- The default **packet-length** is 1518 bytes.
- The default **direction** is ingress.

Command Modes

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

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

Command History

Release	Modification
6.2(1)	This command was added.
7.0(1)	This command was modified to include the following keywords: **type asp-drop**, **type isakmp**, **type raw-data**, and **type webvpn**.
7.0(8)	Added the all option to capture all packets that the ASA drops.
7.2(1)	This command was modified to include the following options: **trace** *trace_count,* **match** *prot,* **real-time, host** *ip,* **any**, *mask,* and *operator.*
8.0(2)	This command was modified to update the path to capture contents.
8.4(1)	The new type keywords ikev1 and ikev2 were added.
8.4(2)	Additional detail was added to the output for IDS.
8.4(4.1)	The **asa_dataplane** option was added to support traffic over the backplane to the ASA CX module.
9.0(1)	The **cluster, cluster exec,** and **reinject-hide** keywords were added. The new **type** option **lacp** was added. Support for multiple-context mode was added for ISAKMP.
9.1(3)	Supports filtering of packets captured on the ASA CX backplane with the **asa_dataplane** option.
9.2(1)	The **asa_dataplane** option was extended to support the ASA FirePOWER module.

Release	Modification
9.3(1)	The **inline-tag** *tag* keyword-argument pair was added to support the SGT plus Ethernet Tagging feature.
9.6(2)	Packet capture of **type asp-drop** supports ACL and match filtering.
9.7(1)	Added the **stop** keyword to manually stop and start the packet capture.
9.8(1)	This command was updated to store the contents of all the active captures to files on flash or disks at the time of box crash.
9.9(1)	Support for capturing clustering persistent tracing and decrypted packets. New options were added: **persist** and **include-decrypted**. In addition, the **ethernet-type ipx** was removed, because IPX corresponds to 3 separate ethernet-types. Instead, use the hexadecimal value of the IPX type you want to capture.
9.10(1)	Added the any4 and any6 keywords for the **match** option to capture IPv4 and IPv6 network traffic respectively.
9.12(1)	Added **cp-cluster** to capture control packets on cluster interface.
9.18(1)	Included **real-time** keyword to enable real-time switch packet capture.
9.20(1)	The **direction** keyword was added to capture switch traffic that flows in **egress**, **ingress**, or **both** directions. This keyword is applicable only for Secure Firewall 4200 model devices.

Usage Guidelines

Capturing packets is useful when troubleshooting connectivity problems or monitoring suspicious activity. You can create multiple captures. The **capture** command is not saved to the running configuration, and is not copied to the standby unit during failover.

The ASA is capable of tracking all IP traffic that flows across it and of capturing all the IP traffic that is destined to it, including all the management traffic (such as SSH and Telnet traffic).

The ASA architecture consists of three different sets of processors for packet processing; this architecture poses certain restrictions on the capability of the capture feature. Typically most of the packet forwarding functionality in the ASA is handled by the two front-end network processors, and packets are sent to the control-plane general-purpose processor only if they need application inspection. The packets are sent to the session management path network processor only if there is a session miss in the accelerated path processor.

Because all the packets that are forwarded or dropped by the ASA hits the two front-end network processors, the packet capture feature is implemented in these network processors. So all the packets that hit the ASA can be captured by these front end processors, if an appropriate capture is configured for those traffic interfaces. On the ingress side, the packets are captured the moment the packet hits the ASA interfaces, and on the egress side the packets are captured just before they are sent out on the wire.

Note Enabling WebVPN capture affects the performance of the ASA. Be sure to disable the capture after you generate the capture files that you need for troubleshooting.

Save the Capture

The contents of any active capture on ASA are saved when the box crashes.

When you activate captures as part of the troubleshooting process, you must note the following points:

- The size of capture buffer to use and if there is enough space on flash/disk.
- The capture buffer should be marked as circular for all the use cases, so that captured packets are the most recent before crash.

The name of the file for saving contents of an active capture is in the format of:

[<context_name>.]<capture_name>.pcap

The *context_name* indicates the name of the user context in which capture is activated in the multi-context mode. For the single context mode, the *context_name* is not applicable.

The *capture_name* indicates the name of the capture that is activated.

The capture save happens before the console or crash dump. This increases the crash downtime by about 5 seconds for a 33 MB capture buffer. The risk of a nested crash is minimal because copying the captured contents to a file is a simple process.

View the Capture

- To view the packet capture at the CLI, use the **show capture** *name* command.
- To save the capture to a file, use the **copy capture** command.
- To see the packet capture information with a web browser, use the **https://***ASA-ip-address***/admin/capture/***capture_name*[**/pcap**] command.

  You are prompted for a username and password. See the **username** command to add a username to the local database.

  If you specify the **pcap** keyword, then a libpcap-format file is downloaded to the web browser and can be saved using the web browser. (A libcap file can be viewed with TCPDUMP or Ethereal.)

If you copy the buffer contents to a TFTP server in ASCII format, you will see only the headers, not the details and hexadecimal dump of the packets. To see the details and hexadecimal dump, you need to transfer the buffer in PCAP format and read it with TCPDUMP or Ethereal.

Stop and Start the Capture

The packets can be stopped from being captured without removing them from the buffer. The stopped status of the capture is displayed. The captured packet is retained in the buffer.

Use the following command to manually stop packet capture:

capture *name* **stop**

Use the following command to start capturing packets:

no capture *name* **stop**

Delete the Capture

Entering **no capture** without any keywords deletes the capture. To preserve the capture, specify the **access-list** or **interface** keyword; the capture is detached from the specified ACL or interface and the capture is preserved.

Real Time Operations

You cannot perform any operations on a capture while the real-time display is in progress. Using the **real-time** keyword with a slow console connection may result in an excessive number of non-displayed packets because of performance considerations. The fixed limit of the buffer is 1000 packets. If the buffer fills up, a counter is maintained of the captured packets. If you open another session, you can disable the real-time display be entering the **no capture real-time** command.

Clustering

You can precede the **capture** command with **cluster exec** to issue the **capture** command on one unit and run the command in all the other units at the same time. After you have performed cluster-wide capture, to copy the same capture file from all units in the cluster at the same time to a TFTP server, enter the **cluster exec copy** command on the master unit.

```
ciscoasa# cluster exec capture
capture_name arguments
ciscoasa# cluster exec copy
 /pcap capture
: cap_name
 tftp
://location
/path
/filename
.pcap
```

Multiple PCAP files, one from each unit, are copied to the TFTP server. The destination capture file name is automatically attached with the unit name, such as filename_A.pcap, filename_B.pcap, and so on. In this example, A and B are cluster unit names.

When you capture traces on cluster units, they are persistent on each cluster node until you manually clear them from the buffer. Decrypted IPsec packets are captured once they enter ASA. The captured packet includes both normal and decapsulated traffic.

Note A different destination name is generated if you add the unit name at the end of the filename.

Limitations

The following are some of the limitations of the capture feature. Most of the limitations are caused by the distributed nature of the ASA architecture and by the hardware accelerators that are being used in the ASA.

- You can configure captures on the cluster control link within a context; only the packet that is associated with the context sent in the cluster control link is captured.
- For a shared VLAN, the following guidelines apply:
    - You can only configure one capture for the VLAN; if you configure a capture in multiple contexts on the shared VLAN, then only the last capture that was configured is used.

  - If you remove the last-configured (active) capture, no captures become active, even if you have previously configured a capture in another context; you must remove and readd the capture to make it active.
  - All traffic that enters the interface to which the capture is attached (and that matches the capture access list) is captured, including traffic to other contexts on the shared VLAN.
  - Therefore, if you enable a capture in Context A for a VLAN that is also used by Context B, both Context A and Context B ingress traffic are captured.

- For egress traffic, only the traffic of the context with the active capture is captured. The only exception is when you do not enable the ICMP inspection (therefore the ICMP traffic does not have a session in the accelerated path). In this case, both ingress and egress ICMP traffic for all contexts on the shared VLAN is captured.
- Configuring a capture typically involves configuring an access list that matches the traffic that needs to be captured. After an access list that matches the traffic pattern is configured, then you need to define a capture and associate this access list to the capture, along with the interface on which the capture needs to be configured. Note that a capture only works if an access list and an interface are associated with a capture for capturing IPv4 traffic. The access list is not required for IPv6 traffic.
- For the ASA CX module traffic, captured packets contain an additional AFBP header that your PCAP viewer might not understand; be sure to use the appropriate plugin to view these packets.
- For inline SGT tagged packets, captured packets contain an additional CMD header that your PCAP viewer might not understand.
- If there is no ingress interface and therefore no global interface, packets sent on the backplane are treated as control packets in the system context. These packets bypass the access list check and are always captured. This behavior applies in both single mode and multiple context mode.
- The **show capture** command shows the correct reason when capturing a specific asp-drop. However, the **show capture** command does not show the correct reason when capturing all asp-drops.

Examples

To capture a packet, enter the following command:

```
ciscoasa# capture captest interface inside
ciscoasa# capture captest interface outside
```

On a web browser, you can view the content of the **capture** command that was issued, named “captest,” at the following location:

```
https://171.69.38.95/admin/capture/captest
```

To download a libpcap file (that web browsers use) to a local machine, enter the following command:

```
https://171.69.38.95/capture/http/pcap
```

The following example shows how to capture a packet in the single-mode when the ASA box crashes:

```
ciscoasa# capture 123 interface inside
```

The contents of capture ‘123’ is saved as *123.pcap* file.

The following example shows how to capture a packet in the multi-mode when the ASA box crashes:

```
ciscoasa# capture 456 interface inside
```

The contents of capture ‘456’ in ‘admin’ context is saved as *admin.456.pcap* file.

The following example shows that the traffic is captured from an outside host at 171.71.69.234 to an inside HTTP server:

```
ciscoasa# access-list http permit tcp host 10.120.56.15 eq http host 171.71.69.234
ciscoasa# access-list http permit tcp host 171.71.69.234 host 10.120.56.15 eq http
ciscoasa# capture http access-list http packet-length 74 interface inside
```

The following example shows how to capture ARP packets:

```
ciscoasa# capture arp ethernet-type arp interface outside
```

The following example inserts five tracer packets into the data stream, where *access-list 101* defines traffic that matches TCP protocol FTP:

```
hostname# capture ftptrace interface outside access-list 101 trace 5
```

To view the traced packets and information about packet processing in an easily readable manner, use the **show capture ftptrace** command.

The following example shows how to display captured packets in real-time:

```
ciscoasa# capture test interface outside real-time
Warning: Using this option with a slow console connection may result in an excess amount
of non-displayed packets due to performance limitations.
Use ctrl-c to terminate real-time capture.
10 packets displayed
12 packets not displayed due to performance limitations
```

The following example shows how to configure an extended access list that matches the IPv4 traffic that needs to be captured:

```
ciscoasa (config)# access-list capture extended permit ip any any
```

The following examples shows how to configure the capture:

```
ciscoasa (config)# capture name access-list acl_name interface interface_name
```

By default, configuring a capture creates a linear capture buffer of size 512 KB. You can optionally configure a circular buffer. By default, only 68 bytes of the packets are captured in the buffer. You can optionally change this value.

The following example creates a capture called “ip-capture” using the capture access list previously configured that is applied to the outside interface:

```
ciscoasa (config)# capture ip-capture access-list capture interface outside
```

The following example creates a capture called “switch-capture” on outside interface for Secure Firewall 3100:

```
ciscoasa (config)# capture switch-capture switch interface outside drop ?
exec mode commands/options:
  disable Disable capturing dropped packets from switch
  mac-filter To capture switch mac-filter drop
ciscoasa(config)# capture switch-capture switch interface outside drop mac-filter
```

The following example shows how to view the capture:

```
ciscoasa (config)# show capture name
```

The following example shows how to end the capture, but retain the buffer:

```
ciscoasa (config)# no capture name access-list acl_name interface interface_name
```

The following example shows how to end the capture and delete the buffer:

```
ciscoasa (config)# no capture name
```

The following example shows how to filter traffic captured on the backplane in single mode:

```
ciscoasa# capture x interface asa_dataplane access-list any4
ciscoasa# capture y interface asa_dataplane match ip any any
```

Note Control packets are captured in the single mode even though you have specified the access list.

The following examples show how to filter traffic captured on the backplane in multiple context mode:

Usage in user context:

```
ciscoasa (contextA)# capture x interface asa_dataplane access-list any4
ciscoasa (contextA)# capture y interface asa_dataplane match ip any any
```

Usage in system context:

```
ciscoasa# capture z interface asa_dataplane
```

Note In multiple context mode, the **access-list** and **match** options are not available in the system context.

Capture for Clustering

To enable capture on all units in the cluster, you can add the **cluster exec** keywords in front of each of these commands.

The following example shows how to create an LACP capture for the clustering environment:

```
ciscoasa (config)# capture lacp type lacp interface gigabitEthernet0/0
```

The following example shows how to create a capture for control path packets in the clustering link:

```
ciscoasa (config)# cap cp interface cluster match udp any eq 49495 any
ciscoasa (config)# cap cp interface cluster match udp any any eq 49495
```

The following example shows how to create a capture for data path packets in the clustering link:

```
ciscoasa (config)# access-list cc1 extended permit udp any any eq 4193
ciscoasa (config)# access-list cc1 extended permit udp any eq 4193 any
ciscoasa (config)# capture dp interface cluster access-list ccl
```

The following example shows how to capture data path traffic through the cluster:

```
ciscoasa (config)# capture abc interface inside match tcp host 1.1.1.1 host 2.2.2.2 eq www
ciscoasa (config)# capture abc interface inside match dup host 1.1.1.1 any
ciscoasa (config)# capture abc interface inside access-list xxx
```

The following example shows how to capture logical update messages for flows that match the real source to the real destination, and capture packets forwarded over CCL that match the real source to the real destination:

```
ciscoasa (config)# access-list dp permit
 real src real dst
```

The following example shows how to capture a certain type of data plane message, such as icmp echo request/response, that is forwarded from one ASA to another ASA using the **match** keyword or the access list for the message type:

```
ciscoasa (config)# capture capture_name interface cluster access-list match icmp any any
```

The following example shows how to create a capture by using access list 103 on a cluster control link in a clustering environment:

```
ciscoasa (config)# access-list 103 permit ip A B
ciscoasa (config)# capture example1 interface cluster
```

In the previous example, if A and B are IP addresses for the CCL interface, only the packets that are sent between these two units are captured.

If A and B are IP addresses for through-device traffic, then the following is true:

- Forwarded packets are captured as usual, provided the source and destination IP addresses are matched with the access list.
- The data path logic update message is captured provided it is for the flow between A and B or for an access list (for example, access-list 103). The capture matches the five-tuple of the embedded flow.
- Although the source and destination addresses in the UDP packet are CCL addresses, if this packet is to update a flow that is associated with addresses A and B, it is also captured. That is, as long as addresses A and B that are embedded in the packet are matched, it is also captured.

The following example shows how to configure capture with persistent option:

```
cluster2-asa5585a(config)# cluster exec capture test interface outside trace persist
  a(LOCAL):*************************************************************
  cluster2-asa5585a(config)#
```

Now, you can send some traffic.

```
cluster2-asa5585a(config)# cluster exec show packet-tracer

  a(LOCAL):*************************************************************
  tracer 29/25 (allocate/freed), handle 29/25 (allocated/freed), error 0
  =======  Tracer origin-id a:23, hop 0 =======
  packet-id: Protocol: 0 src-port: 0 dst-port: 0
  Phase: 1
  Type: CAPTURE
  Subtype:
  Result: ALLOW
  Config:
  Additional Information:
  MAC Access list
  Phase: 2
  Type: ACCESS-LIST
  Subtype:
  Result: DROP
  Config:
  Implicit Rule
  Additional Information:
  MAC Access list
  Result:
  input-interface: outside
  input-status: up
  input-line-status: up
  Action: drop
  Drop-reason: (l2_acl) FP L2 rule drop
```

The following example shows that, to free up some memory you must clear the captured persistent traces from the box.

```
ciscoasa# cluster exec clear packet-trace
```

The following example displays how to configure the capture with include-decrypted option:

```
  cluster2-asa5585a(config)# cluster exec show capture

  a(LOCAL):*************************************************************
  capture in type raw-data trace interface outside include-decrypted [Capturing - 588 bytes]

  capture out type raw-data trace interface outside include-decrypted [Capturing - 420
bytes]
  cluster2-asa5585a(config)#
```

Now, you can send some ICMP traffic through IPSec tunnel. The capture command obtains the decrypted ICMP packets as outlined:

```
  cluster2-asa5585a(config)# cluster exec show capture in | i icmp
  a(LOCAL):*************************************************************
  b:********************************************************************
  cluster2-asa5585a(config)# cluster exec show capture out | i icmp
  a(LOCAL):*************************************************************
  b:********************************************************************
  cluster2-asa5585a(config)# cluster exec show capture in | i icmp
  a(LOCAL):*************************************************************
     8: 07:22:57.065014       802.1Q vlan#212 P0 211.1.1.1 > 213.1.1.2: icmp: echo request

  b:********************************************************************
  cluster2-asa5585a(config)# cluster exec show capture out | i icmp
  a(LOCAL):*************************************************************
    10: 07:22:57.068004       802.1Q vlan#214 P0 213.1.1.2 > 211.1.1.1: icmp: echo reply
  b:********************************************************************
  cluster2-asa5585a(config)#
```

The following example shows how to create and start an egress traffic capture for a switch:

```
ciscoasa(config)# capture switch_cap switch interface gigabitEthernet0/0 direction ?
exec mode commands/options:
  both     To capture switch bi-directional traffic
  egress   To capture switch egressing traffic
  ingress  To capture switch ingressing traffic

ciscoasa(config)# capture switch_cap switch interface gigabitEthernet0/0 direction egress
ciscoasa(config)# no capture switch_cap switch stop
```

Related Commands

Command	Description
clear capture	Clears the capture buffer.
copy capture	Copies a capture file to a server.
show capture	Displays the capture configuration when no options are specified.

cd

To change the current working directory to the one specified, use the **cd** command in privileged EXEC mode.

cd [**disk0:** | **disk1:** | **flash:**] [*path* **]**

Syntax Description

disk0:	Specifies the internal Flash memory, followed by a colon.
disk1:	Specifies the removable, external Flash memory card, followed by a colon.
flash:	Specifies the internal Flash memory, followed by a colon. In the ASA 5500 series, the **flash** keyword is aliased to **disk0**.
path	(Optional) The absolute path of the directory to change to.

Command Default

If you do not specify a directory, the directory is changed to the root directory.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Examples

The following example shows how to change to the “config” directory:

```
ciscoasa# cd flash:/config/
```

Related Commands

Command	Description
pwd	Displays the current working directory.

cdp-url

To specify the CDP to be included in certificates issued by the local CA, use the **cdp-url** command in ca server configuration mode. To revert to the default CDP, use the **no** form of this command.

[**no**] **cdp-url** *url*

Syntax Description

url	Specifies the URL where a validating party obtains revocation status for certificates issued by the local CA. The URL must be less than 500 alphanumeric characters. **Note** ASA supports both IPv4 and IPv6 CDP URLs. Enclose IPv6 addresses in square brackets, for example: *http://**[**0:0:0:0:0.18:0a01:7c16**]***.

Command Default

(For ASA versions 9.12(1) and earlier) The default CDP URL is that of the ASA that includes the local CA. The default URL is in the format: http://hostname.domain/+CSCOCA+/asa_ca.crl.

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 server configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.
9.20(1)	Support for IPv6 CDP URL was added.

Usage Guidelines

The CDP is an extension that can be included in issued certificates to specify the location where a validating party can obtain revocation status for the certificate. Only one CDP can be configured at a time.

Note If a CDP URL is specified, it is the responsibility of the administrator to maintain access to the current CRL from that location.

Examples

The following example (applicable only for ASA versions 9.12(1) and earlier) configures a CDP at 10.10.10.12 for certificates issued by the local CA server:

```
ciscoasa(config)# crypto ca server
ciscoasa
(config-ca-server)
# cdp-url http://10.10.10.12/ca/crl
```

```
ciscoasa
(config-ca-server)
#
```

The following example configures a CDP IPv6 url:

```
ciscoasa(config)# crypto ca server
ciscoasa
(config-ca-server)
# cdp-url http://[0:0:0:0:0:ffff:0a01:7c16]
ciscoasa
(config-ca-server)
#
```

Related Commands

Command	Description
crypto ca server	Provides access to ca server configuration mode CLI command set, which allows you to configure and manage a local CA.
crypto ca server crl issue	Forces the issuance of a CRL.
crypto ca server revoke	Marks a certificate issued by a local CA server as revoked in the certificate database and CRL.
crypto ca server unrevoke	Unrevokes a previously revoked certificate issued by a local CA server.
lifetime crl	Specifies the lifetime of the certificate revocation list.

certificate

To add the indicated certificate, use the **certificate** command in crypto ca certificate chain configuration mode. To delete the certificate, use the **no** form of this command.

certificate [**ca** | **ra-encrypt** | **ra-sign** | **ra-general**] *certificate-serial-number*
no certificate *certificate-serial-number*

Syntax Description

ca	Indicates that the certificate is a CA issuing certificate.
certificate-serial-number	Specifies the serial number of the certificate in hexadecimal format ending with the word "quit."
ra-encrypt	Indicates that the certificate is an RA key encipherment certificate used in SCEP.
ra-general	Indicates that the certificate is an RA certificate used for digital signing and key encipherment in SCEP messaging.
ra-sign	Indicates that the certificate is an RA digital signature certificate used in SCEP messaging.

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 certificate chain configuration	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

When this command is issued, the ASA interprets the data included with it as the certificate in hexadecimal format. A **quit** string indicates the end of the certificate.

A CA is an authority in a network that issues and manages security credentials and public key for message encryption. As part of a public key infrastructure, a CA checks with a RA to verify information provided by the requester of a digital certificate. If the RA verifies the requester information, the CA can then issue a certificate.

Examples

The following example adds a CA certificate with the serial number 29573D5FF010FE25B45:

```
ciscoasa
(config)#
crypto ca trustpoint central
ciscoasa
(ca-trustpoint)#
crypto ca certificate chain central
ciscoasa
(ca-cert-chain)#
certificate ca 29573D5FF010FE25B45
  30820345 308202EF A0030201 02021029 572A3FF2 96EF854F D0D6732F E25B4530
  0D06092A 864886F7 0D010105 05003081 8F311630 1406092A 864886F7 0D010901
  16076140 622E636F 6D310B30 09060355 04061302 55533116 30140603 55040813
  0D6D6173 73616368 75736574 74733111 300F0603 55040713 08667261 6E6B6C69
  6E310E30 0C060355 040A1305 63697363 6F310F30 0D060355 040B1306 726F6F74
  6F75311C 301A0603 55040313 136D732D 726F6F74 2D736861 2D30362D 32303031
  301E170D 30313036 32363134 31313430 5A170D32 32303630 34313430 3133305A
  30818F31 16301406 092A8648 86F70D01 09011607 6140622E 636F6D31 0B300906
  03550406 13025553 31163014 06035504 08130D6D 61737361 63687573 65747473
  3111300F 06035504 07130866 72616E6B 6C696E31 0E300C06 0355040A 13056369
  73636F31 0F300D06 0355040B 1306726F 6F746F75 311C301A 06035504 0313136D
  732D726F 6F742D73 68612D30 362D3230 3031305C 300D0609 2A864886 F70D0101
  01050003 4B003048 024100AA 3EB9859B 8670A6FB 5E7D2223 5C11BCFE 48E6D3A8
  181643ED CF7E75EE E77D83DF 26E51876 97D8281E 9F58E4B0 353FDA41 29FC791B
  1E14219C 847D19F4 A51B7B02 03010001 A3820123 3082011F 300B0603 551D0F04
  04030201 C6300F06 03551D13 0101FF04 05300301 01FF301D 0603551D 0E041604
  14E0D412 3ACC96C2 FBF651F3 3F66C0CE A62AB63B 323081CD 0603551D 1F0481C5
  3081C230 3EA03CA0 3A86386C 6461703A 2F2F7732 6B616476 616E6365 64737276
  2F436572 74456E72 6F6C6C2F 6D732D72 6F6F742D 7368612D 30362D32 3030312E
  63726C30 3EA03CA0 3A863868 7474703A 2F2F7732 6B616476 616E6365 64737276
  2F436572 74456E72 6F6C6C2F 6D732D72 6F6F742D 7368612D 30362D32 3030312E
  63726C30 40A03EA0 3C863A66 696C653A 2F2F5C5C 77326B61 6476616E 63656473
  72765C43 65727445 6E726F6C 6C5C6D73 2D726F6F 742D7368 612D3036 2D323030
  312E6372 6C301006 092B0601 04018237 15010403 02010130 0D06092A 864886F7
  0D010105 05000341 0056221E 03F377B9 E6900BF7 BCB3568E ADBA146F 3B8A71F3
  DF9EB96C BB1873B2 B6268B7C 0229D8D0 FFB40433 C8B3CB41 0E4D212B 2AEECD77
  BEA3C1FE 5EE2AB6D 91
  quit
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.
crypto ca certificate chain	Enters certificate crypto ca certificate chain mode.
crypto ca trustpoint	Enters ca trustpoint mode.
show running-config crypto map	Displays all configuration for all the crypto maps.

certificate-group-map

To associate a rule entry from a certificate map with a tunnel group, use the **certificate-group-map** command in webvpn configuration mode. To clear current tunnel-group map associations, use the **no** form of this command.

certificate-group-map *certificate_map_name index tunnel_group_name*
no certificate-group-map

Syntax Description

certificate_map_name	The name of a certificate map.
index	The numeric identifier for a map entry in the certificate map. The index value can be in the range of 1-65535.
tunnel_group_name	The name of the tunnel group chosen if the map entry matches the certificate. The *tunnel-group name* must already exist.

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
Webvpn configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.

Usage Guidelines

With the **certificate-group-map** command in effect, if a certificate received from a WebVPN client corresponds to a map entry, the resulting tunnel group is associated with the connection, overriding any tunnel group choice made by the user.

Multiple instances of the **certificate-group-map** command allow multiple mappings.

Examples

The following example shows how to associate rule 6 for a tunnel group named tgl:

ciscoasa (config)# webvpn

```
hostname(config-webvpn)# certificate-group-map map1 6 tg1
hostname(config-webvpn)#
```

Related Commands

Command	Description
crypto ca certificate map	Enters ca certificate map configuration mode for configuring rules based on the certificate issuer and subject distinguished names (DNs).
tunnel-group-map	Configures the policy and rules by which certificate-based IKE sessions are mapped to tunnel groups.

chain

To enable sending a certificate chain, use the **chain** command in tunnel-group ipsec-attributes configuration mode. To return this command to the default, use the **no** form of this command.

chain
no chain

Syntax Description

This command has no arguments or keywords.

Command Default

The default setting for this command is disabled.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

You can apply this attribute to all IPsec tunnel group types.

Entering this command includes the root certificate and any subordinate CA certificates in the transmission.

Examples

The following example entered in tunnel-group-ipsec attributes configuration mode, enables sending a chain for an IPSec LAN-to-LAN tunnel group with the IP address of 209.165.200.225, which includes the root certificate and any subordinate CA certificates:

```
ciscoasa(config)# tunnel-group 209.165.200.225 type IPSec_L2L
ciscoasa(config)# tunnel-group 209.165.200.225 ipsec-attributes
ciscoasa(config-tunnel-ipsec)# chain
ciscoasa(config-tunnel-ipsec)#
```

Related Commands

Command	Description
clear-configure tunnel-group	Clears all configured tunnel groups.
show running-config tunnel-group	Shows the current tunnel group configuration.
tunnel-group ipsec-attributes	Configures the tunnel-group ipsec-attributes for this group.

change-password

To enable users to change their own account passwords, use the **change-password** command in privileged EXEC mode.

change-password [**/silent**] [**old-password** *old-password* [**new-password** *new-password*]]

Syntax Description

new-password *new-password*	Specifies the new password.
old-password *old-password*	Reauthenticates the user.

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
Privileged EXEC	• Yes	• Yes	—	—	• Yes
Global configuration	• Yes	• Yes	• Yes	—	• Yes

Command History

Release	Modification
8.4(4.1)	This command was added.

Usage Guidelines

If users omit the passwords, the ASA prompts them for input. When users enter the **change-password** command, they are asked to save their running configuration. After a user has successfully changed the password, a message appears to remind the user to save configuration changes.

Examples

The following example changes a user account password:

```
ciscoasa# change-password old-password
myoldpassword000
 new password
mynewpassword123
```

Related Commands

Command	Description
show run password-policy	Shows the password policy for the current context.
clear configure password-policy	Resets password policy for the current context to the default value.

Command	Description
clear configure username	Removes a username from a user account.

changeto

To change between security contexts and the system, use the **changeto** command in privileged EXEC mode.

changeto { **system** | **context** *name* }

Syntax Description

context *name*	Changes to the context with the specified name.
system	Changes to the system execution space.

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
Privileged EXEC	• Yes	• Yes	—	• Yes	• Yes

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

If you log into the system execution space or the admin context, you can change between contexts and perform configuration and monitoring tasks within each context. The “running” configuration that you edit in configuration mode, or that is used in the **copy** or **write** commands, depends on which execution space you are in. When you are in the system execution space, the running configuration consists only of the system configuration; when you are in a context execution space, the running configuration consists only of that context. For example, you cannot view all running configurations (system plus all contexts) by entering the **show running-config** command. Only the current configuration appears.

Examples

The following example changes between contexts and the system in privileged EXEC mode:

```
ciscoasa/admin# changeto system
ciscoasa# changeto context customerA
ciscoasa/customerA#
```

The following example changes between the system and the admin context in interface configuration mode. When you change between execution spaces, and you are in a configuration mode, the mode changes to the global configuration mode in the new execution space.

```
ciscoasa(config-if)# changeto context admin
ciscoasa/admin(config)#
```

Related Commands

Command	Description
admin-context	Sets a context to be the admin context.
context	Creates a security context in the system configuration and enters context configuration mode.
show context	Shows a list of contexts (system execution space) or information about the current context.

channel-group

To assign a physical interface to an EtherChannel, use the **channel-group** command in interface configuration mode. To unassaign the interface, use the **no** form of this command.

channel-group *channel_id* **mode** { **active** | **passive** | **on** } [**vss-id** { **1** | **2** }]
no channel-group *channel_id*

Syntax Description

channel_id	Specifies the EtherChannel to which you want to assign this interface, between 1 and 48.
vss-id {**1** | **2**}	(Optional) With clustering, if you are connecting the ASA to two switches in a VSS or vPC, then configure the **vss-id** keyword to identify to which switch this interface is connected (1 or 2). You must also use the **port-channel span-cluster vss-load-balance** command for the port-channel interface.
mode {**active** | **passive** | **on**}	You can configure each physical interface in an EtherChannel to be: • Active—Sends and receives Link Aggregation Control Protocol (LACP) updates. An active EtherChannel can establish connectivity with either an active or a passive EtherChannel. You should use the active mode unless you need to minimize the amount of LACP traffic. • Passive—Receives LACP updates. A passive EtherChannel can only establish connectivity with an active EtherChannel. • On—The EtherChannel is always on, and LACP is not used. An “on” EtherChannel can only establish a connection with another “on” EtherChannel.

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
Interface configuration	• Yes	• Yes	• Yes	—	• Yes

Command History

Release	Modification
8.4(1)	We added this command.
9.0(1)	We added the **vss-id** keyword to support ASA clustering and spanned EtherChannels.

Usage Guidelines

Each channel group can have eight active interfaces. Note that you can assign up to 16 interfaces to a channel group. While only eight interfaces can be active, the remaining interfaces can act as standby links in case of interface failure.

All interfaces in the channel group must be the same type and speed. The first interface added to the channel group determines the correct type and speed.

If the port-channel interface for this channel ID does not yet exist in the configuration, one will be added:

```
interface port-channel
 channel_id
```

The Link Aggregation Control Protocol (LACP) aggregates interfaces by exchanging the Link Aggregation Control Protocol Data Units (LACPDUs) between two network devices. LACP coordinates the automatic addition and deletion of links to the EtherChannel without user intervention. It also handles misconfigurations and checks that both ends of member interfaces are connected to the correct channel group. “On” mode cannot use standby interfaces in the channel group when an interface goes down, and the connectivity and configurations are not checked.

ASA Clustering

You can include multiple interfaces per ASA in a spanned EtherChannel. Multiple interfaces per ASA are especially useful for connecting to both switches in a VSS or vPC. If you are connecting the ASA to two switches in a VSS or vPC, then you should enable VSS load balancing by using the **vss-load-balance** keyword. This feature ensures that the physical link connections between the ASAs to the VSS (or vPC) pair are balanced. You must configure the **vss-id** keyword in the **channel-group** command for each member interface before enabling load balancing.

Examples

The following example assigns interfaces to channel group 1:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# channel-group 1 mode active
ciscoasa(config-if)# interface GigabitEthernet0/1
ciscoasa(config-if)# channel-group 1 mode active
ciscoasa(config-if)# interface GigabitEthernet0/2
ciscoasa(config-if)# channel-group 1 mode passive
```

Related Commands

Command	Description
channel-group	Adds an interface to an EtherChannel.
interface port-channel	Configures an EtherChannel.
lacp max-bundle	Specifies the maximum number of active interfaces allowed in the channel group.
lacp port-priority	Sets the priority for a physical interface in the channel group.
lacp system-priority	Sets the LACP system priority.
port-channel load-balance	Configures the load-balancing algorithm.
port-channel min-bundle	Specifies the minimum number of active interfaces required for the port-channel interface to become active.

Command	Description
show lacp	Displays LACP information such as traffic statistics, system identifier and neighbor details.
show port-channel	Displays EtherChannel information in a detailed and one-line summary form. This command also displays the port and port-channel information.
show port-channel load-balance	Displays port-channel load-balance information along with the hash result and member interface selected for a given set of parameters.

character-encoding

To specify the global character encoding in WebVPN portal pages, use the **character-encoding** command in webvpn configuration mode. To remove the value of the character-encoding attribute, use the **no** form of this command.

character-encoding *charset*
no character-encoding *charset*

Syntax Description

charset	String consisting of up to 40 characters, and equal to one of the valid character sets identified in http://www.iana.org/assignments/character-sets . You can use either the name or the alias of a character set listed on that page. Examples include iso-8859-1, shift_jis, and ibm850. The string is case-insensitive. The command interpreter converts upper case to lower case in the ASA configuration.

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
Webvpn configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.1(1)	This command was added.

Usage Guidelines

Character encoding , also called “character coding” and “a character set,” is the pairing of raw data (such as 0s and 1s) and characters to represent the data. The language determines the character encoding method to use. Some languages use the same method, while others do not. Usually, the geographic region determines the default encoding method used by the browser, but the user can change this. The browser can also detect the encoding specified on the page, and render the document accordingly. The character-encoding attribute lets the user specify the value of the character-encoding method into the WebVPN portal page to ensure that the browser renders it correctly, regardless of the region in which the user is using the browser, or any changes made to the browser.

The character-encoding attribute is a global setting that, by default, all WebVPN portal pages inherit. However, the user can override the file-encoding attribute for Common Internet File System (CIFS) servers that use character encoding that differs from the value of the character-encoding attribute. Use different file-encoding values for CIFS servers that require different character encodings.

The WebVPN portal pages downloaded from the CIFS server to the WebVPN user encode the value of the WebVPN file-encoding attribute identifying the server, or if one does not, they inherit the value of the character-encoding attribute. The remote user browser maps this value to an entry in its character encoding set to determine the proper character set to use. The WebVPN portal pages do not specify a value if WebVPN configuration does not specify a file-encoding entry for the CIFS server and the character-encoding attribute is not set. The remote browser uses its own default encoding if the WebVPN portal page does not specify the character encoding or if it specifies a character encoding value that the browser does not support.

The mapping of CIFS servers to their appropriate character encoding, globally with the webvpn character-encoding attribute, and individually with file-encoding overrides, provides for the accurate handling and display of CIFS pages when the correct rendering of file names or directory paths, as well as pages, is an issue.

Note The character-encoding and file-encoding values do not exclude the font family to be used by the browser. The user needs to complement the setting of one these values with the **page style** command in webvpn customization command mode to replace the font family if you are using Japanese Shift_JIS character encoding, as shown in the following example, or enter the **no page style** command in webvpn customization command mode to remove the font family.

The encoding type set on the remote browser determines the character set for WebVPN portal pages when this attribute does not have a value.

Examples

The following example sets the character-encoding attribute to support Japanese Shift_JIS characters, removes the font family, and retains the default background color:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# character-encoding shift_jis
ciscoasa(config-webvpn)# customization DfltCustomization
ciscoasa(config-webvpn-custom)# page style background-color:white
ciscoasa(config-webvpn-custom)#
```

Related Commands

Command	Description
debug webvpn cifs	Displays debugging messages about the CIFS server.
file-encoding	Specifies CIFS servers and associated character encoding to override the value of this attribute.
show running-config [**all**] webvpn	Displays the running configuration for WebVPN. Use the **all** keyword to include the default configuration.

checkheaps

To configure checkheaps verification intervals, use the **checkheaps** command in global configuration mode. To set the value to the default, use the **no** form of this command.

checkheaps { **check-interval** | **validate-checksum** } *seconds*
no checkheaps { **check-interval** | **validate-checksum** } [*seconds*]

Syntax Description

check-interval	Sets the buffer verification interval. The buffer verification process checks the sanity of the heap (allocated and freed memory buffers). During each invocation of the process, the ASA checks the entire heap, validating each memory buffer. If there is a discrepancy, the ASA issues either an "allocated buffer error" or a "free buffer error." If there is an error, the ASA dumps traceback information when possible and reloads.
seconds	Sets the interval in seconds between 1 and 2147483.
validate-checksum	Sets the code space checksum validation interval. When the ASA first boots up, the ASA calculates a hash of the entire code. Later, during the periodic check, the ASA generates a new hash and compares it to the original. If there is a mismatch, the ASA issues a "text checksum checkheaps error." If there is an error, the ASA dumps traceback information when possible and reloads.

Command Default

The default intervals are 60 seconds each.

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

Checkheaps is a periodic process that verifies the sanity of the heap memory buffers (dynamic memory is allocated from the system heap memory region) and the integrity of the code region.

Examples

The following example sets the buffer allocation interval to 200 seconds and the code space checksum interval to 500 seconds:

```
ciscoasa(config)# checkheaps check-interval 200
ciscoasa(config)# checkheaps validate-checksum 500
```

Related Commands

Command	Description
show checkheaps	Shows checkheaps statistics.

check-retransmission

To prevent against TCP retransmission style attacks, use the **check-retransmission** command in tcp-map configuration mode. To remove this specification, use the **no** form of this command.

check-retransmission
no check-retransmission

Syntax Description

This command has no arguments or keywords.

Command Default

The default is disabled.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

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

Use the **tcp-map** command to enter tcp-map configuration mode. To prevent against TCP retransmission style attacks that arise from end-system interpretation of inconsistent retransmissions, use the **check-retransmission** command in tcp-map configuration mode.

The ASA will make efforts to verify if the data in retransmits are the same as the original. If the data does not match, then the connection is dropped by the ASA. When this feature is enabled, packets on the TCP connection are only allowed in order. For more details, see the **queue-limit** command.

Examples

The following example enables the TCP check-retransmission feature on all TCP flows:

```
ciscoasa(config)# access-list TCP extended permit tcp any any
ciscoasa(config)# tcp-map tmap
ciscoasa(config-tcp-map)# check-retransmission
ciscoasa(config)# class-map cmap
ciscoasa(config-cmap)# match access-list TCP
ciscoasa(config)# policy-map pmap
ciscoasa(config-pmap)# class cmap
ciscoasa(config-pmap)# set connection advanced-options tmap
ciscoasa(config)# service-policy pmap global
```

Related Commands

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

checksum-verification

To enable or disable TCP checksum verification, use the **checksum-verification** command in tcp-map configuration mode. To remove this specification, use the **no** form of this command.

checksum-verification
no checksum-verification

Syntax Description

This command has no arguments or keywords.

Command Default

Checksum verification 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
Tcp-map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

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

Use the **tcp-map** command to enter tcp-map configuration mode. Use the **checksum-verification** command in tcp-map configuration mode to enable TCP checksum verification. If the check fails, the packet is dropped.

Examples

The following example enables TCP checksum verification on TCP connections from 10.0.0.0 to 20.0.0.0:

```
ciscoasa(config)# access-list TCP1 extended permit tcp 10.0.0.0 255.0.0.0 20.0.0.0 255.0.0.0
ciscoasa(config)# tcp-map tmap
ciscoasa(config-tcp-map)# checksum-verification
ciscoasa(config)# class-map cmap
ciscoasa(config-cmap)# match access-list TCP1
ciscoasa(config)# policy-map pmap
ciscoasa(config-pmap)# class cmap
ciscoasa(config-pmap)# set connection advanced-options tmap
ciscoasa(config)# service-policy pmap global
```

Related Commands

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

checksum-verification

To enable or disable TCP checksum verification, use the **checksum-verification** command in tcp-map configuration mode. To remove this specification, use the **no** form of this command.

checksum-verification
no checksum-verification

Syntax Description

This command has no arguments or keywords.

Command Default

Checksum verification 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
Tcp-map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

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

Use the **tcp-map** command to enter tcp-map configuration mode. Use the **checksum-verification** command in tcp-map configuration mode to enable TCP checksum verification. If the check fails, the packet is dropped.

Examples

The following example enables TCP checksum verification on TCP connections from 10.0.0.0 to 20.0.0.0:

```
ciscoasa(config)# access-list TCP1 extended permit tcp 10.0.0.0 255.0.0.0 20.0.0.0 255.0.0.0
ciscoasa(config)# tcp-map tmap
ciscoasa(config-tcp-map)# checksum-verification
ciscoasa(config)# class-map cmap
ciscoasa(config-cmap)# match access-list TCP1
ciscoasa(config)# policy-map pmap
ciscoasa(config-pmap)# class cmap
ciscoasa(config-pmap)# set connection advanced-options tmap
ciscoasa(config)# service-policy pmap global
```

Related Commands

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

cipc security-mode authenticated (Deprecated)

To force Cisco IP Communicator (CIPC) softphones to operate in authenticated mode when CIPC softphones are deployed in a voice and data VLAN scenario, use the **cipc security-mode authenticated** command in phone-proxy configuration mode. To turn off this command when CIPC softphones support encryption, use the **no** form of this command.

cipc security-mode authenticated
no cipc security-mode authenticated

Syntax Description

This command has no arguments or keywords.

Command Default

Be default, this command is disabled via the no form of the command.

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
Phone-proxy configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(4)	The command was added.
9.4(1)	This command was deprecated along with all **phone-proxy** mode commands.

Usage Guidelines

Separating voice and data traffic by using VLANs is a security best practice to hide voice streams from security threats that attempt to penetrate the data VLAN. However, Cisco IP Communicator (CIPC) softphone applications must connect to their respective IP phones, which reside on the voice VLAN. This requirement makes segregating voice and data VLANs an issue because the SIP and SCCP protocols dynamically negotiate the RTP/RTCP ports on a wide range of ports. This dynamic negotiation requires that a range of ports be open between the two VLANs.

Note Earlier versions of CIPC that do not support Authenticated mode are not supported with the Phone Proxy.

To allow CIPC softphones on the data VLAN to connect to their respective IP phones on the voice VLAN without requiring access between the VLANs on a wide range of ports, you can configure the Phone Proxy with the **cipc security-mode authenticated** command.

This command allows the Phone Proxy to look for CIPC configuration files and force CIPC softphones to be in authenticated mode rather than encrypted mode, because current versions of CIPC do not support encrypted mode.

When this command is enabled, the Phone Proxy parses the phones configuration file to determine if the phone is a CIPC softphone and changes the security mode to authenticated. Additionally, CIPC softphones support authenticated mode only while the Phone Proxy, by default, forces all phones to be in encrypted mode.

Examples

The following example shows the use of the **cipc security-mode authenticated** command to force Cisco IP Communicator (CIPC) softphones to operate in authenticated mode when CIPC softphones are deployed in a voice and data VLAN scenario:

```
ciscoasa
(config)# phone-proxy asa_phone_proxy
ciscoasa(config-phone-proxy)#cipc security-mode authenticated
```

Related Commands

Command	Description
phone-proxy	Configures the Phone Proxy instance.

clacp static-port-priority

To disable dynamic port priority in LACP for a clustering spanned EtherChannel, which is required for more than 8 active EtherChannel members, use the **clacp static-port-priority** command in global configuration mode. To enable dynamic port priority, use the **no** form of this command.

Note Supported on ASA hardware models only.

clacp static-port-priority
no clacp static-port-priority

Syntax Description

This command has no arguments or keywords.

Command Default

This command is disabled by default; dynamic port priority is enabled.

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
9.2(1)	We added this command.

Usage Guidelines

Some switches do not support dynamic port priority, so this command improves switch compatibility. Moreover, it enables support of more than 8 active spanned EtherChannel members, up to 32 members. Without this command, only 8 active members and 8 standby members are supported.

ASA EtherChannels support up to 16 active links. With *spanned* EtherChannels, that functionality is extended to support up to 32 active links across the cluster when used with two switches in a vPC and when you disable dynamic port priority with the **clacp static-port-priority** command. The switches must support EtherChannels with 16 active links, for example, the Nexus 7000 with with F2-Series 10 Gigabit Ethernet Module.

For switches in a VSS or vPC that support 8 active links, you can configure 16 active links in the spanned EtherChannel (8 connected to each switch).

Note If you want to use more than 8 active links in a spanned EtherChannel, you cannot also have standby links; the support for 9 to 32 active links requires you to disable cLACP dynamic port priority that allows the use of standby links.

Examples

The following example disables dynamic port priority:

```
ciscoasa(config)# clacp static-port-priority
```

Related Commands

Command	Description
clacp system-mac	Sets the cLACP system ID.

clacp system-mac

To manually configure the cLACP system ID on the master unit in an ASA cluster, use the **clacp system-mac** command in cluster group configuration mode. To retsore the default setting, use the **no** form of this command.

clacp system-mac { *mac_address* | **auto** } [**system-priority** *number*]
no clacp system-mac { *mac_address* | **auto** } [**system-priority** *number*]

Syntax Description

mac_address	Manually sets the system ID in the form *H .H .H* , where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0A-00-00-AA-AA is entered as 000A.0000.AAAA.
auto	Auto-generates the system ID.
system-priority *number*	Sets the system priority, between 1 and 65535. The priority is used to decide which unit is in charge of making a bundling decision. By default, the ASA uses priority 1, which is the highest priority. The priority needs to be higher than the priority on the switch.

Command Default

By default, the system-mac is auto-generated (**auto**).

By default, the system-priority is 1.

Command Modes

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

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

Command History

Release	Modification
9.0(1)	We added this command.

Usage Guidelines

When using spanned EtherChannels, the ASA uses cLACP to negotiate the EtherChannel with the neighbor switch. ASAs in a cluster collaborate in cLACP negotiation so that they appear as a single (virtual) device to the switch. One parameter in cLACP negotiation is a system ID, which is in the format of a MAC address. All ASAs use the same system ID: auto-generated by the master unit (the default) and replicated to all slaves; or manually specified in this command. You might want to manually configure the MAC address for troubleshooting purposes, for example, so you can use an easily identified MAC address. Typically, you would use the auto-generated MAC address.

This command is not part of the bootstrap configuration, and is replicated from the master unit to the slave units. However, you cannot change this value after you enable clustering.

Examples

The following example manually configures a system ID:

```
cluster group pod1
local-unit unit1
cluster-interface port-channel1 ip 192.168.1.1 255.255.255.0
priority 1
key chuntheunavoidable
health-check
clacp system-mac 000a.0000.aaaa
enable noconfirm
```

Related Commands

Command	Description
cluster group	Configures cluster parameters.

class (global)

To create a resource class to which to assign a security context, use the **class** command in global configuration mode. To remove a class, use the **no** form of this command.

class *name*
no class *name*

Syntax Description

name	Specifies the name as a string up to 20 characters long. To set the limits for the default class, enter **default** for the name.

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
7.2(1)	This command was added.

Usage Guidelines

By default, all security contexts have unlimited access to the resources of the ASA, except where maximum limits per context are enforced. However, if you find that one or more contexts use too many resources, and they cause other contexts to be denied connections, for example, then you can configure resource management to limit the use of resources per context.

The ASA manages resources by assigning contexts to resource classes. Each context uses the resource limits set by the class.

When you create a class, the ASA does not set aside a portion of the resources for each context assigned to the class; rather, the ASA sets the maximum limit for a context. If you oversubscribe resources, or allow some resources to be unlimited, a few contexts can “use up” those resources, potentially affecting service to other contexts. See the **limit-resource** command to set the resources for the class.

All contexts belong to the default class if they are not assigned to another class; you do not have to actively assign a context to the default class.

If a context belongs to a class other than the default class, those class settings always override the default class settings. However, if the other class has any settings that are not defined, then the member context uses the default class for those limits. For example, if you create a class with a 2 percent limit for all concurrent connections, but no other limits, then all other limits are inherited from the default class. Conversely, if you create a class with limits for all resources, the class uses no settings from the default class.

By default, the default class provides unlimited access to resources for all contexts, except for the following limits, which are by default set to the maximum allowed per context:

- Telnet sessions—5 sessions.
- SSH sessions—5 sessions.
- MAC addresses—65,535 entries.

Examples

The following example sets the default class limit for conns to 10 percent instead of unlimited:

```
ciscoasa(config)# class default
ciscoasa(config-class)# limit-resource conns 10%
```

All other resources remain at unlimited.

To add a class called gold, enter the following commands:

```
ciscoasa(config)# class gold
ciscoasa(config-class)#
limit-resource mac-addresses 10000
ciscoasa(config-class)#
limit-resource conns 15%
ciscoasa(config-class)#
limit-resource rate conns 1000
ciscoasa(config-class)#
limit-resource rate inspects 500
ciscoasa(config-class)#
limit-resource hosts 9000
ciscoasa(config-class)#
limit-resource asdm 5
ciscoasa(config-class)#
limit-resource ssh 5
ciscoasa(config-class)#
limit-resource rate syslogs 5000
ciscoasa(config-class)#
limit-resource telnet 5
ciscoasa(config-class)#
limit-resource xlates 36000
ciscoasa(config-class)#
limit-resource routes 5000
```

Related Commands

Command	Description
clear configure class	Clears the class configuration.
context	Configures a security context.
limit-resource	Sets the resource limit for a class.
member	Assigns a context to a resource class.
show class	Shows the contexts assigned to a class.

class (policy-map)

To assign a class map to a policy map where you can assign actions to the class map traffic, use the **class** command in policy-map configuration mode. To remove a class map from a policy map, use the **no** form of this command.

class *classmap_name*
no class *classmap_name*

Syntax Description

classmap_name	Specifies the name for the class map. For a Layer 3/4 policy map (the **policy-map** command), you must specify a Layer 3/4 class map name (the **class-map** or **class-map type management** command). For an inspection policy map (the **policy-map type inspect** command), you must specify an inspection class map name (the **class-map type inspect** 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
Policy-map configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

To use the **class** command, use the Modular Policy Framework. To use a class in a Layer 3/4 policy map, enter the following commands:

1. **class-map**—Identify the traffic on which you want to perform actions.
2. **policy-map**—Identify the actions associated with each class map.
   a. **class**—Identify the class map on which you want to perform actions.
   b. *commands for supported features* —For a given class map, you can configure many actions for various features, including QoS, application inspection, CSC or AIP SSM, TCP and UDP connections limits and timeout, and TCP normalization. See the CLI configuration guide for more details about the commands available for each feature.
3. **service-policy**—Assigns the policy map to an interface or globally.

To use a class in an inspection policy map, enter the following commands:

1. **class-map type inspect**—Identify the traffic on which you want to perform actions.
2. **policy-map type inspect**—Identify the actions associated with each class map.
   a. **class**—Identify the inspection class map on which you want to perform actions.
   b. *commands for application types* —See the CLI configuration guide for commands available for each application type. Actions supported in class configuration mode of an inspection policy map include:
   c. Dropping a packet
   d. Dropping a connection
   e. Resetting a connection
   f. Logging
   g. Rate-limiting of messages
   h. Masking content
   i. **parameters**—Configure parameters that affect the inspection engine. The CLI enters parameters configuration mode. See the CLI configuration guide for available commands.
3. **class-map**—Identify the traffic on which you want to perform actions.
4. **policy-map**—Identify the actions associated with each class map.
   a. **class**—Identify the Layer 3/4 class map on which you want to perform actions.
   b. **inspect** *application inspect_policy_map* —Enables application inspection, and calls an inspection policy map to perform special actions.
5. **service-policy**—Assigns the policy map to an interface or globally.

The configuration always includes a class map called **class-default** that matches all traffic. At the end of every Layer 3/4 policy map, the configuration includes the **class-default** class map with no actions defined. You can optionally use this class map when you want to match all traffic, and do not want to bother creating another class map. In fact, some features are only configurable for the **class-default** class map, such as the **shape** command.

Including the **class-default** class map, up to 63 **class** and **match** commands can be configured in a policy map.

Examples

The following is an example of a **policy-map** command for connection policy that includes the **class** command. It limits the number of connections allowed to the web server 10.1.1.1:

```
ciscoasa(config)# access-list http-server permit tcp any host 10.1.1.1
ciscoasa(config)# class-map http-server
ciscoasa(config-cmap)# match access-list http-server
ciscoasa(config)# policy-map global-policy
ciscoasa(config-pmap)# description This policy map defines a policy concerning connection
to http server.
ciscoasa(config-pmap)# class http-server
ciscoasa(config-pmap-c)# set connection conn-max 256
```

The following example shows how multi-match works in a policy map:

```
ciscoasa(config)# class-map inspection_default
ciscoasa(config-cmap)# match default-inspection-traffic
ciscoasa(config)# class-map http_traffic
ciscoasa(config-cmap)# match port tcp eq 80
ciscoasa(config)# policy-map outside_policy
ciscoasa(config-pmap)# class inspection_default
ciscoasa(config-pmap-c)# inspect http http_map
ciscoasa(config-pmap-c)# inspect sip
ciscoasa(config-pmap)# class http_traffic
ciscoasa(config-pmap-c)# set connection timeout tcp 0:10:0
```

The following example shows how traffic matches the first available class map, and will not match any subsequent class maps that specify actions in the same feature domain:

```
ciscoasa(config)# class-map telnet_traffic
ciscoasa(config-cmap)# match port tcp eq 23
ciscoasa(config)# class-map ftp_traffic
ciscoasa(config-cmap)# match port tcp eq 21
ciscoasa(config)# class-map tcp_traffic
ciscoasa(config-cmap)# match port tcp range 1 65535
ciscoasa(config)# class-map udp_traffic
ciscoasa(config-cmap)# match port udp range 0 65535
ciscoasa(config)# policy-map global_policy
ciscoasa(config-pmap)# class telnet_traffic
ciscoasa(config-pmap-c)# set connection timeout tcp 0:0:0
ciscoasa(config-pmap-c)# set connection conn-max 100
ciscoasa(config-pmap)# class ftp_traffic
ciscoasa(config-pmap-c)# set connection timeout tcp 0:5:0
ciscoasa(config-pmap-c)# set connection conn-max 50
ciscoasa(config-pmap)# class tcp_traffic
ciscoasa(config-pmap-c)# set connection timeout tcp 2:0:0
ciscoasa(config-pmap-c)# set connection conn-max 2000
```

When a Telnet connection is initiated, it matches **class telnet_traffic**. Similarly, if an FTP connection is initiated, it matches **class ftp_traffic**. For any TCP connection other than Telnet and FTP, it will match **class tcp_traffic**. Even though a Telnet or FTP connection can match **class tcp_traffic**, the ASA does not make this match because they previously matched other classes.

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map.
class-map type management	Creates a Layer 3/4 class map for management traffic.
clear configure policy-map	Removes all policy map configuration, except for any policy map that is in use in a **service-policy** command.
match	Defines the traffic-matching parameters.
policy-map	Configures a policy; that is, an association of one or more traffic classes, each with one or more actions.

class-map

When using the Modular Policy Framework, identify Layer 3 or 4 traffic to which you want to apply actions by using the **class-map** command (without the **type** keyword) in global configuration mode. To delete a class map, use the **no** form of this command.

class-map *class_map_name*
no class-map *class_map_name*

Syntax Description

class_map_name	Specifies the class map name up to 40 characters in length. The names “class-default” and any name that begins with “_internal” or “_default” are reserved. All types of class maps use the same name space, so you cannot resuse a name already used by another type of class map.

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

This type of class map is for Layer 3/4 through traffic only. For management traffic destined to the ASA, see the **class-map type management** command.

A Layer 3/4 class map identifies Layer 3 and 4 traffic to which you want to apply actions. You can create multiple Layer 3/4 class maps for each Layer 3/4 policy map.

Default Class Maps

The configuration includes a default Layer 3/4 class map that the ASA uses in the default global policy. It is called **inspection_default** and matches the default inspection traffic:

```
class-map inspection_default
 match default-inspection-traffic
```

Another class map that exists in the default configuration is called class-default, and it matches all traffic:

```
class-map class-default
 match any
```

This class map appears at the end of all Layer 3/4 policy maps and essentially tells the ASA to not perform any actions on all other traffic. You can use the class-default class map if desired, rather than making your own **match any** class map. In fact, some features are only available for class-default, such as QoS traffic shaping.

Maximum Class Maps

The maximum number of class maps of all types is 255 in single mode or per context in multiple mode. Class maps include the following types:

- **class-map**
- **class-map type management**
- **class-map type inspection**
- **class-map type regex**
- **match** commands in policy-map type inspect configuration mode

This limit also includes default class maps of all types.

Configuration Overview

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

Use the **class-map** command to enter class-map configuration mode. From class-map configuration mode, you can define the traffic to include in the class using the **match** command. A Layer 3/4 class map contains, at most, one **match** command (with the exception of the **matchtunnel-group** and **matchdefault-inspection-traffic** commands) that identifies the traffic included in the class map.

Examples

The following example creates four Layer 3/4 class maps:

```
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 all_http
ciscoasa(config-cmap)# description "This class-map matches all HTTP traffic"
ciscoasa(config-cmap)# match port tcp eq http
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 type management	Creates a class map for traffic to the ASA.
policy-map	Creates a policy map by associating the traffic class with one or more actions.
policy-map type inspect	Defines special actions for application inspection.
service-policy	Creates a security policy by associating the policy map with one or more interfaces.
show running-config class-map	Displays the information about the class map configuration.

class-map type inspect

When using the Modular Policy Framework, match criteria that is specific to an inspection application by using the **class-map type inspect** command in global configuration mode. To delete an inspection class map, use the **no** form of this command.

class-map type inspect *application* [**match-all** | **match-any**] *class_map_name*
class-map [**type inspect** *application* [**match-all** | **match-any**]] *class_map_name*

Syntax Description

application	Specifies the type of application traffic you want to match. Available types include: • **dcerpc** • **diameter** • **dns** • **ftp** • **h323** • **http** • **im** • **rtsp** • **scansafe** • **sip**
class_map_name	Specifies the class map name up to 40 characters in length. The names "class-default" and any name that begins with "_internal" or "_default" are reserved. All types of class maps use the same name space, so you cannot resuse a name already used by another type of class map.
match-all	(Optional) Specifies that traffic must match all criteria to match the class map. **match-all** is the default if you do not specify an option.
match-any	(Optional) Specifies that traffic can match one or more criteria to match the class map.

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
7.2(1)	This command was added.
8.0(2)	The **match-any** keyword was added.
9.0(1)	The **scansafe** keyword was added.
9.5(2)	The **dcerpc** and **diameter** keywords were added.

Usage Guidelines

Modular Policy Framework lets you configure special actions for many application inspections. When you enable an inspection engine in the Layer 3/4 policy map, you can also optionally enable actions as defined in an *inspection policy map* (see the **policy-map type inspect** command).

In the inspection policy map, you can identify the traffic you want to act upon by creating an inspection class map. The class map contains one or more **match** commands. (You can alternatively use **match** commands directly in the inspection policy map if you want to pair a single criterion with an action). You can match criteria that is specific to an application. For example, for DNS traffic, you can match the domain name in a DNS query.

A class map groups multiple traffic matches (in a match-all class map), or lets you match any of a list of matches (in a match-any class map). The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you group multiple match commands, and you can reuse class maps. For the traffic that you identify in this class map, you can specify actions such as dropping, resetting, and/or logging the connection in the inspection policy map.

The maximum number of class maps of all types is 255 in single mode or per context in multiple mode. Class maps include the following types:

- **class-map**
- **class-map type management**
- **class-map type inspection**
- **class-map type regex**
- **match** commands in policy-map type inspect configuration mode

This limit also includes default class maps of all types. See the **class-map** command for more information.

Examples

The following example creates an HTTP class map that must match all criteria:

```
ciscoasa(config-cmap)# class-map type inspect http match-all http-traffic
ciscoasa(config-cmap)# match req-resp content-type mismatch
ciscoasa(config-cmap)# match request body length gt 1000
ciscoasa(config-cmap)# match not request uri regex class URLs
```

The following example creates an HTTP class map that can match any of the criteria:

```
ciscoasa(config-cmap)# class-map type inspect http match-any monitor-http
ciscoasa(config-cmap)# match request method get
ciscoasa(config-cmap)# match request method put
ciscoasa(config-cmap)# match request method post
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map for through traffic.
policy-map	Creates a policy map by associating the traffic class with one or more actions.
policy-map type inspect	Defines special actions for application inspection.
service-policy	Creates a security policy by associating the policy map with one or more interfaces.
show running-config class-map	Displays the information about the class map configuration.

class-map type management

When using the Modular Policy Framework, identify Layer 3 or 4 management traffic destined for the ASA to which you want to apply actions by using the **class-map type management** command in global configuration mode. To delete a class map, use the **no** form of this command.

class-map type management *class_map_name*
no class-map type management *class_map_name*

Syntax Description

class_map_name	Specifies the class map name up to 40 characters in length. The names "class-default" and any name that begins with "_internal" or "_default" are reserved. All types of class maps use the same name space, so you cannot resuse a name already used by another type of class map.

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
7.2(1)	This command was added.
8.0(2)	The s**et connection** command is now available for a Layer 3/4 management class map, for to-the-ASA management traffic. Only the **conn-max** and **embryonic-conn-max** keywords are available.

Usage Guidelines

This type of class map is for management traffic only. For through traffic, see the **class-map** command (without the **type** keyword).

For management traffic to the ASA, you might want to perform actions specific to this kind of traffic. The types of actions available for a management class map in the policy map are specialized for management traffic. For example, this type of class map lets you inspect RADIUS accounting traffic and set connection limits.

A Layer 3/4 class map identifies Layer 3 and 4 traffic to which you want to apply actions. The maximum number of class maps of all types is 255 in single mode or per context in multiple mode.

You can create multiple Layer 3/4 class maps (management or through traffic) for each Layer 3/4 policy map.

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

Use the **class-map type management** command to enter class-map configuration mode. From class-map configuration mode, you can define the traffic to include in the class using the **match** command. You can specify a management class map that can match an access list or TCP or UDP ports. A Layer 3/4 class map contains, at most, one **match** command that identifies the traffic included in the class map.

The maximum number of class maps of all types is 255 in single mode or per context in multiple mode. Class maps include the following types:

- **class-map**
- **class-map type management**
- **class-map type inspection**
- **class-map type regex**
- **match** commands in policy-map type inspect configuration mode

This limit also includes default class maps of all types. See the **class-map** command for more information.

Examples

The following example creates a Layer 3/4 management class map:

```
ciscoasa(config)# class-map type management radius_acct
ciscoasa(config-cmap)# match port tcp eq 10000
```

Related Commands

Command	Description
class-map	Creates a Layer 3/4 class map for through traffic.
policy-map	Creates a policy map by associating the traffic class with one or more actions.
policy-map type inspect	Defines special actions for application inspection.
service-policy	Creates a security policy by associating the policy map with one or more interfaces.
show running-config class-map	Displays the information about the class map configuration.

class-map type regex

When using the Modular Policy Framework, group regular expressions for use with matching text by using the **class-map type regex** command in global configuration mode. To delete a regular expression class map, use the **no** form of this command.

class-map type managementclass_map_name *class_map_name*
no class-map [**type regex match-any**] *class_map_name*

Syntax Description

class_map_name	Specifies the class map name up to 40 characters in length. The names “class-default” and any name that begins with “_internal” or “_default” are reserved. All types of class maps use the same name space, so you cannot resuse a name already used by another type of class map.
match-any	Specifies that the traffic matches the class map if it matches only one of the regular expressions. **match-any** is the only option.

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
7.2(1)	This command was added.

Usage Guidelines

Modular Policy Framework lets you configure special actions for many application inspections. When you enable an inspection engine in the Layer 3/4 policy map, you can also optionally enable actions as defined in an *inspection policy map* (see the **policy-map type inspect** command).

In the inspection policy map, you can identify the traffic you want to act upon by creating an inspection class map containing one or more **match** commands or you can use **match** commands directly in the inspection policy map. Some **match** commands let you identify text in a packet using a regular expression; for example, you can match URL strings inside HTTP packets. You can group regular expressions in a regular expression class map.

Before you create a regular expression class map, create the regular expressions using the **regex** command. Then, identify the named regular expressions in class-map configuration mode using the **match regex** command.

The maximum number of class maps of all types is 255 in single mode or per context in multiple mode. Class maps include the following types:

- **class-map**
- **class-map type management**
- **class-map type inspection**
- **class-map type regex**
- **match** commands in policy-map type inspect configuration mode

This limit also includes default class maps of all types. See the **class-map** command for more information.

Examples

The following example creates two regular expressions, and adds them to a regular expression class map. Traffic matches the class map if it includes the string "example.com" or "example2.com."

```
ciscoasa(config)# regex url_example example\.com
ciscoasa(config)# regex url_example2 example2\.com
ciscoasa(config)# class-map type regex match-any URLs
ciscoasa(config-cmap)# match
 regex
 url_example
ciscoasa(config-cmap)# match
 regex
 url_example2
```

Related Commands

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