CHAPTER 8

client-access-rule through crl configure Commands

client-access-rule

To configure rules that limit the remote access client types and versions that can connect via IPSec through the security appliance, use the **client-access-rule** command in group-policy configuration mode. To delete a rule, use the **no** form of this command.

To delete all rules, use the **no client-access-rule command** with only the priority argument. This deletes all configured rules, including a null rule created by issuing the **client-access-rule none** command.

When there are no client access rules, users inherit any rules that exist in the default group policy. To prevent users from inheriting client access rules, use the **client-access-rule none** command. The result of doing so is that all client types and versions can connect.

client-access-rule *priority* **{permit | deny} type** *type* **version** *version* **| none**

no client-access-rule *priority* [**{permit | deny} type** *type* **version** *version*]

Syntax Description

deny	Denies connections for devices of a particular type and/or version.
none	Allows no client access rules. Sets client-access-rule to a null value, thereby allowing no restriction. Prevents inheriting a value from a default or specified group policy.
permit	Permits connections for devices of a particular type and/or version.
priority	Determines the priority of the rule. The rule with the lowest integer has the highest priority. Therefore, the rule with the lowest integer that matches a client type and/or version is the rule that applies. If a lower priority rule contradicts, the security appliance ignores it.
type *type*	Identifies device types via free-form strings, for example VPN 3002. A string must match exactly its appearance in the **show vpn-sessiondb remote** display, except that you can use the * character as a wildcard.
version *version*	Identifies the device version via free-form strings, for example 7.0(1). A string must match exactly its appearance in the **show vpn-sessiondb remote** display, except that you can use the * character as a wildcard.

Defaults

By default, there are no access rules.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Construct rules according to these caveats:

- If you do not define any rules, the security appliance permits all connection types.
- When a client matches none of the rules, the security appliance denies the connection. This means that if you define a deny rule, you must also define at least one permit rule, or the security appliance denies all connections.
- For both software and hardware clients, type and version must match exactly their appearance in the **show vpn-sessiondb remote** display.
- The * character is a wildcard, which you can use multiple times in each rule. For example, **client-access-rule 3 deny type * version 3.*** creates a priority 3 client access rule that denies all client types running release versions 3.x software.
- You can construct a maximum of 25 rules per group policy.
- There is a limit of 255 characters for an entire set of rules.
- You can use n/a for clients that do not send client type and/or version.

Examples

The following example shows how to create client access rules for the group policy named FirstGroup. These rules permit VPN Clients running software version 4.1, while denying all VPN 3002 hardware clients:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# client-access-rule 1 d t VPN3002 v *
hostname(config-group-policy)# client-access-rule 2 p * v 4.1
```

client-firewall

To set personal firewall policies that the security appliance pushes to the VPN Client during IKE tunnel negotiation, use the **client-firewall** command in group-policy configuration mode. To delete a firewall policy, use the **no** form of this command.

When there are no firewall policies, users inherit any that exist in the default or other group policy. To prevent users from inheriting such firewall policies, use the **client-firewall none** command.

client-firewall none

client-firewall {**opt** | **req**} **custom vendor-id** *num* **product-id** *num* **policy** {**AYT** | **CPP acl-in** *acl* **acl-out** *acl*} [**description** *string*]

client-firewall {**opt** | **req**} **zonelabs-zonealarm policy** {**AYT** | **CPP acl-in** *acl* **acl-out** *acl*}

client-firewall {**opt** | **req**} **zonelabs-zonealarmorpro policy** {**AYT** | **CPP acl-in** *acl* **acl-out** *acl*}

client-firewall {**opt** | **req**} **zonelabs-zonealarmpro policy** {**AYT** | **CPP acl-in** *acl* **acl-out** *acl*}

client-firewall {**opt** | **req**} **cisco-integrated acl-in** *acl* **acl-out** *acl*}

client-firewall {**opt** | **req**} **sygate-personal**

client-firewall {**opt** | **req**} **sygate-personal-pro**

client-firewall {**opt** | **req**} **sygate-personal-agent**

client-firewall {**opt** | **req**} **networkice-blackice**

client-firewall {**opt** | **req**} **cisco-security-agent**

Syntax Description

acl-in <*acl*>	Provides the policy the client uses for inbound traffic
acl-out <*acl*>	Provides the policy the client uses for outbound traffic
AYT	Specifies that the client PC firewall application controls the firewall policy. The security appliance checks to make sure the firewall is running. It asks, “Are You There?” If there is no response, the security appliance tears down the tunnel.
cisco-integrated	Specifies Cisco Integrated firewall type.
cisco-security-agent	Specifies Cisco Intrusion Prevention Security Agent firewall type
CPP	Specifies Policy Pushed as source of the VPN Client firewall policy
custom	Specifies Custom firewall type.
description <*string*>	Describes the firewall.
networkice-blackice	Specifies Network ICE Black ICE firewall type
none	Indicates that there is no client firewall policy. Sets a firewall policy with a null value, thereby disallowing one. Prevents inheriting a firewall policy from a default or specified group policy.
opt	Indicates an optional firewall type.
product-id	Identifies the firewall product
req	Indicates a required firewall type

sygate-personal	Specifies Sygate Personal firewall type
sygate-personal-pro	Specifies Sygate Personal Pro firewall type
sygate-security-agent	Specifies Sygate Security Agent firewall type
vendor-id	Identifies the firewall vendor
zonelabs-zonealarm	Specifies Zone Labs Zone Alarm firewall type
zonelabs-zonealarmorpro policy	Specifies Zone Labs Zone Alarm or Pro firewall type
zonelabs-zonealarmpro policy	Specifies Zone Labs Zone Alarm Pro firewall type

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Only one instance of this command can be configured.

Examples

The following example shows how to set a client firewall policy that requires Cisco Intrusion Prevention Security Agent for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# client-firewall req cisco-security-agent
```

client-update

To issue a client-update for all active remote clients on all tunnel-groups or for a particular tunnel group, use the **client-update** command in privileged EXEC mode. To configure and change client-update parameters at the global level, use the **client-update** command in global configuration mode. To configure and change client-update tunnel-group IPSec-attributes parameters, use the **client-update** command in tunnel-group ipsec-attributes configuration mode. If the client is already running a software version on the list of revision numbers, it does not need to update its software. If the client is not running a software version on the list, it should update. You can specify up to three client update entries, one for each client type.

To disable a client update, use the **no** form of this command.

Global configuration mode commands:

client-update {**enable** | **type** *type* **url** *url-string* **rev-nums** *rev-nums*}

no client-update {**enable** | **type** *type* **url** *url-string* **rev-nums** *rev-nums*}

Tunnel-group ipsec-attributes mode commands:

client-update type *type* **url** *url-string* **rev-nums** *rev-nums*

no client-update type *type* **url** *url-string* **rev-nums** *rev-nums*

Privileged EXEC mode commands:

client-update {**all** | *tunnel-group*}

no client-update *tunnel-group*

Syntax Description

all	(Available only in privileged EXEC mode.) Applies the action to all active remote clients in all tunnel groups. You cannot use the keyword **all** with the **no** form of the command.
enable	(Available only in global configuration mode). Enables remote client software updates.
rev-nums *rev-nums*	(Not available in privileged EXEC mode.) Specifies the software or firmware images for this client. Enter up to 4, in any order, separated by commas.
tunnel-group	(Available only in privileged EXEC mode.) Specifies the name of a valid tunnel-group for remote client update.

type *type*	(Not available in privileged EXEC mode.) Specifies the operating systems to notify of a client update. The list of operating systems comprises the following: • Windows: all windows-based platforms • WIN9X: Windows 95, Windows 98, and Windows ME platforms • WinNT: Windows NT 4.0, Windows 2000, and Windows XP platforms • vpn3002: VPN 3002 hardware client
url *url-string*	(Not available in privileged EXEC mode.) Specifies the URL for the software/firmware image. This URL must point to a file appropriate for this client.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—
Global configuration	•	—	•	—	—
Tunnel-group ipsec-attributes configuration	•	—	•	—	—

Command History

Release	Modification
7.0(1)	This command was introduced.
7.1(1)	Added tunnel-group ipsec-attributes configuration mode.

Usage Guidelines

In tunnel-group ipsec-attributes configuration mode, you can apply this attribute only to the IPSec remote-access tunnel-group type.

The **client-update** command lets you enable the update; specify the types and revision numbers of clients to which the update applies; provide a URL or IP address from which to get the update; and, in the case of Windows clients, optionally notify users that they should update their VPN client version. For Windows clients, you can provide a mechanism for users to accomplish that update. For VPN 3002 Hardware Client users, the update occurs automatically, with no notification.

To configure the client-update mechanism, do the following steps:

Step 1 In global configuration mode, enable client update by entering the command:

```
hostname(config)# client-update enable
hostname(config)#
```

Step 2 In global configuration mode, configure the parameters for the client update that you want to apply to all clients of a particular type. That is, specify the type of client and the URL or IP address from which to get the updated image. In addition, you must specify a revision number. If the user's client revision number matches one of the specified revision numbers, there is no need to update the client. This command configures the client-update parameters for all clients of the specified type across the entire security appliance. For example:

```
hostname(config)# client-update type windows url https://support/updates/ rev-nums 4.6.1
hostname(config)#
```

See the Examples section for an illustration of configuring a tunnel group for a VPN 3002 hardware client.

Note For all Windows clients, you must use the protocol "http://" or "https://" as the prefix for the URL. For the VPN3002 Hardware Client, you must specify protocol "tftp://" instead.

Alternatively, you can configure client update just for individual tunnel-groups, rather than for all clients of a particular type. (See Step 3.)

Note You can have the browser automatically start an application by including the application name at the end of the URL; for example: **https://support/updates/vpnclient.exe.**

Step 3 After you have enabled client update, you can define a set of client-update parameters for a particular ipsec-ra tunnel group. To do this, in tunnel-group ipsec-attributes mode, specify the tunnel-group name and its type, and the URL or IP address from which to get the updated image. In addition, you must specify a revision number. If the user's client revision number matches one of the specified revision numbers, there is no need to update the client; for example, to issue a client update for all Windows clients:

```
hostname(config)# tunnel-group remotegrp type ipsec-ra
hostname(config)# tunnel-group remotegrp ipsec-attributes
hostname(config-tunnel-ipsec)# client-update type windows url https://support/updates/
rev-nums 4.6.1
hostname(config-tunnel-ipsec)#
```

See the Examples section for an illustration of configuring a tunnel group for a VPN 3002 hardware client. VPN 3002 clients update without user intervention, and users receive no notification message.

Step 4 Optionally, you can send a notice to active users with outdated Windows clients that their VPN client needs updating. For these users, a pop-up window appears, offering the opportunity to launch a browser and download the updated software from the site specified in the URL. The only part of this message that you can configure is the URL. (See Step 2 or 3.) Users who are not active get a notification message the next time they log on. You can send this notice to all active clients on all tunnel groups, or you can send it to clients on a particular tunnel group. For example, to notify all active clients on all tunnel groups, you would enter the following command in privileged EXEC mode:

```
hostname# client-update all
hostname#
```

If the user's client revision number matches one of the specified revision numbers, there is no need to update the client, and users receive no notification message. VPN 3002 clients update without user intervention and users receive no notification message.

Note If you specify the client-update type as **windows** (specfying all Windows-based platforms) and later want to enter a client-update type of **win9x** or **winnt** for the same entity, you must first remove the windows client type with the **no** form of the command, then use new client-update commands to specify the new client types.

Examples

The following example, entered in global configuration mode, enables client update for all active remote clients on all tunnel groups:

```
hostname(config)# client-update enable
hostname#
```

The following example applies only to Windows (win9x, winnt, or windows). Entered in global configuration mode, it configures client update parameters for all Windows-based clients. It designates the revision number, 4.7 and the URL for retrieving the update, which is https://support/updates.

```
hostname(config)# client-update type windows url https://support/updates/ rev-nums 4.7
hostname(config)#
```

The following example applies only to VPN 3002 Hardware Clients. Entered in tunnel-group ipsec-attributes configuration mode, it configures client update parameters for the IPSec remote-access tunnel-group “salesgrp”. It designates the revision number, 4.7 and uses the TFTP protocol for retrieving the updated software from the site with the IP address 192.168.1.1:

```
hostname(config)# tunnel-group salesgrp type ipsec-ra
hostname(config)# tunnel-group salesgrp ipsec-attributes
hostname(config-tunnel-ipsec)# client-update type vpn3002 url tftp:192.168.1.1 rev-nums
4.7
hostname(config-tunnel-ipsec)#
```

The following example, entered in privileged EXEC mode, sends a client-update notification to all connected remote clients in the tunnel group named “remotegrp” that need to update their client software. Clients in other groups do not get an update notification:

```
hostname# client-update remotegrp
hostname#
```

Related Commands

Command	Description
clear configure client-update	Clears the entire client-update configuration.
show running-config client-update	Shows the current client-update configuration.
tunnel-group ipsec-attributes	Configures the tunnel-group ipsec-attributes for this group.

clock set

To manually set the clock on the security appliance, use the **clock set** command in privileged EXEC mode.

clock set *hh*:*mm*:*ss* {*month day* | *day month*} *year*

Syntax Description

day	Sets the day of the month, from 1 to 31. You can enter the day and month as **april 1** or as **1 april**, for example, depending on your standard date format.
hh:*mm*:*ss*	Sets the hour, minutes, and seconds in 24-hour time. For example, set **20:54:00** for 8:54 pm.
month	Sets the month. Depending on your standard date format, you can enter the day and month as **april 1** or as **1 april**.
year	Sets the year using four digits, for example, **2004**. The year range is 1993 to 2035.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

If you have not entered any **clock** configuration commands, the default time zone for the **clock set** command is UTC. If you change the time zone after you enter the **clock set** command using the **clock timezone** command, the time automatically adjusts to the new time zone. However, if you enter the **clock set** command after you establish the time zone with the **clock timezone** command, then enter the time appropriate for the new time zone and not for UTC. Similarly, if you enter the **clock summer-time** command after the **clock set** command, the time adjusts for daylight saving. If you enter the **clock set** command after the **clock summer-time** command, enter the correct time for daylight saving.

This command sets the time in the hardware chip, and does not save the time in the configuration file. This time endures reboots. Unlike the other **clock** commands, this command is a privileged EXEC command. To reset the clock, you need to set a new time for the **clock set** command.

Examples

The following example sets the time zone to MST, the daylight saving time to the default period in the U.S., and the current time for MDT to 1:15 p.m. on July 27, 2004:

```
hostname(config)# clock timezone MST -7
hostname(config)# clock summer-time MDT recurring
hostname(config)# exit
hostname# clock set 13:15:0 jul 27 2004
hostname# show clock
13:15:00.652 MDT Tue Jul 27 2004
```

The following example sets the clock to 8:15 on July 27, 2004 in the UTC time zone, and then sets the time zone to MST and the daylight saving time to the default period in the U.S. The end time (1:15 in MDT) is the same as the previous example.

```
hostname# clock set 20:15:0 jul 27 2004
hostname# configure terminal
hostname(config)# clock timezone MST -7
hostname(config)# clock summer-time MDT recurring
hostname# show clock
13:15:00.652 MDT Tue Jul 27 2004
```

Related Commands

Command	Description
clock summer-time	Sets the date range to show daylight saving time.
clock timezone	Sets the time zone.
show clock	Shows the current time.

clock summer-time

To set the date range for daylight saving time for the display of the security appliance time, use the **clock summer-time** command in global configuration mode. To disable the daylight saving time dates, use the **no** form of this command.

clock summer-time *zone* **recurring** [*week weekday month hh*:*mm week weekday month hh*:*mm*] [*offset*]

no clock summer-time [*zone* **recurring** [*week weekday month hh*:*mm week weekday month hh*:*mm*] [*offset*]]

clock summer-time *zone* **date** {*day month* | *month day*} *year hh*:*mm* {*day month* | *month day*} *year hh*:*mm* [*offset*]

no clock summer-time [*zone* **date** {*day month* | *month day*} *year hh*:*mm* {*day month* | *month day*} *year hh*:*mm* [*offset*]]

Syntax Description

date	Specifies the start and end dates for daylight saving time as a specific date in a specific year. If you use this keyword, you need to reset the dates every year.
day	Sets the day of the month, from 1 to 31. You can enter the day and month as **April 1** or as **1 April**, for example, depending on your standard date format.
hh:mm	Sets the hour and minutes in 24-hour time.
month	Sets the month as a string. For the **date** command, you can enter the day and month as **April 1** or as **1 April**, for example, depending on your standard date format.
offset	(Optional) Sets the number of minutes to change the time for daylight saving time. By default, the value is 60 minutes.
recurring	Specifies the start and end dates for daylight saving time, in the form of a day and time of the month, and not a specific date in a year. This keyword lets you set a recurring date range that you do not need to alter yearly. If you do not specify any dates, the security appliance uses the default date range for the United States: 2:00 a.m. on the first Sunday in April to 2:00 a.m. on the last Sunday in October.
week	(Optional) Specifies the week of the month as an integer between 1 and 4 or as the words **first** or **last**. For example, if the day might fall in the partial fifth week, then specify **last**.
weekday	(Optional) Specifies the day of the week: **Monday**, **Tuesday**, **Wednesday**, and so on.
year	Sets the year using four digits, for example, **2004**. The year range is 1993 to 2035.
zone	Specifies the time zone as a string, for example, **PDT** for Pacific Daylight Time. When the security appliance shows the daylight saving time according to the date range you set with this command, the time zone changes to the value you set here. See the **clock timezone** to set the base time zone to a zone other than UTC.

Defaults

The default offset is 60 minutes.

The default recurring date range is from 2:00 a.m. on the first Sunday in April to 2:00 a.m. on the last Sunday in October.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

For the Southern Hemisphere, the security appliance accepts the start month to be later in the year than the end month, for example, from October to March.

Examples

The following example sets the daylight saving date range for Australia:

```
hostname(config)# clock summer-time PDT recurring last Sunday October 2:00 last Sunday
March 2:00
```

Some countries start daylight saving on a specific date. In the following example, daylight saving time is configured to start on April 1, 2004, at 3 a.m. and end on October 1, 2004, at 4 a.m.

```
hostname(config)# clock summer-time UTC date 1 April 2004 3:00 1 October 2004 4:00
```

Related Commands

Command	Description
clock set	Manually sets the clock on the security appliance.
clock timezone	Sets the time zone.
ntp server	Identifies an NTP server.
show clock	Shows the current time.

clock timezone

To set the time zone for the security appliance clock, use the **clock timezone** command in global configuration mode. To set the time zone back to the default of UTC, use the **no** form of this command. The **clock set** command or the time derived from an NTP server sets the time in UTC. You must set the time zone as an offset of UTC using this command.

clock timezone *zone* [**-**]*hours* [*minutes*]

no clock timezone [*zone* [**-**]*hours* [*minutes*]]

Syntax Description

zone	Specifies the time zone as a string, for example, **PST** for Pacific Standard Time.
[**-**]*hours*	Sets the number of hours of offset from UTC. For example, PST is **-8** hours.
minutes	(Optional) Sets the number of minutes of offset from UTC.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To set daylight saving time, see the **clock summer-time** command.

Examples

The following example sets the time zone to Pacific Standard Time, which is -8 hours from UTC:

```
hostname(config)# clock timezone PST -8
```

Related Commands

Command	Description
clock set	Manually sets the clock on the security appliance.
clock summer-time	Sets the date range to show daylight saving time.

Command	Description
ntp server	Identifies an NTP server.
show clock	Shows the current time.

cluster encryption

To enable encryption for messages exchanged on the virtual load-balancing cluster, use the **cluster encryption** command in VPN load-balancing mode. To disable encryption, use the **no** form of this command.

cluster encryption

no cluster encryption

Note VPN load balancing requires an active 3DES/AES license. The security appliance checks for the existence of this crypto license before enabling load balancing. If it does not detect an active 3DES or AES license, the security appliance prevents the enabling of load balancing and also prevents internal configuration of 3DES by the load balancing system unless the license permits this usage.

Syntax Description

This command has no arguments or variables.

Defaults

Encryption is disabled by default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

This command turns encryption on or off for messages exchanged on the virtual load-balancing cluster.

Before configuring the **cluster encryption** command, you must have first used the **vpn load-balancing** command to enter VPN load-balancing mode. You must also use the **cluster key** command to configure the cluster shared-secret key before enabling cluster encryption.

Note When using encryption, you must first configure the command **isakmp enable** *inside*, where *inside* designates the load-balancing inside interface. If isakmp is not enabled on the load-balancing inside interface, you will get an error message when you try to configure cluster encryption.

Examples

The following is an example of a VPN load-balancing command sequence that includes a **cluster encryption** command that enables encryption for the virtual load-balancing cluster:

```
hostname(config)# interface GigabitEthernet 0/1
hostname(config-if)# ip address 209.165.202.159 255.255.255.0
hostname(config)# nameif test
hostname(config)# interface GigabitEthernet 0/2
hostname(config-if)# ip address 209.165.201.30 255.255.255.0
hostname(config)# nameif foo
hostname(config)# vpn load-balancing
hostname(config-load-balancing)# interface lbpublic test
hostname(config-load-balancing)# interface lbprivate foo
hostname(config-load-balancing)# cluster ip address 209.165.202.224
hostname(config-load-balancing)# cluster key 123456789
hostname(config-load-balancing)# cluster encryption
hostname(config-load-balancing)# participate
```

Related Commands

Command	Description
cluster key	Specifies the shared-secret key for the cluster.
vpn load-balancing	Enters VPN load-balancing mode.

cluster ip address

To set the IP address of the virtual load-balancing cluster, use the **cluster ip address** command in VPN load-balancing mode. To remove the IP address specification, use the **no** form of this command.

cluster ip address *ip-address*

no cluster ip address [*ip-address*]

Syntax Description

ip-address	The IP address that you want to assign to the virtual load-balancing cluster.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

You must first use the **vpn load-balancing** command to enter VPN load-balancing mode and configure the interface to which the virtual cluster IP address refers.

The cluster ip address must be on the same subnet as the interface for which you are configuring the virtual cluster.

In the **no** form of the command, if you specify the optional *ip-address* value, it must match the existing cluster IP address before the **no cluster ip address** command can be completed.

Examples

The following is an example of a VPN load-balancing command sequence that includes a **cluster ip address** command that sets the IP address of the virtual load-balancing cluster to 209.165.202.224:

```
hostname(config)# interface GigabitEthernet 0/1
hostname(config-if)# ip address 209.165.202.159 255.255.255.0
hostname(config)# nameif test
hostname(config)# interface GigabitEthernet 0/2
hostname(config-if)# ip address 209.165.201.30 255.255.255.0
hostname(config)# nameif foo
hostname(config)# vpn load-balancing
hostname(config-load-balancing)# interface lbpublic test
hostname(config-load-balancing)# interface lbprivate foo
hostname(config-load-balancing)# cluster ip address 209.165.202.224
```

```
hostname(config-load-balancing)# participate
```

Related Commands

Command	Description
interface	Sets the interfaces of the device.
nameif	Assigns a name to an interface.
vpn load-balancing	Enters VPN load-balancing mode.

cluster key

To set the shared secret for IPSec site-to-site tunnel exchanges on the virtual load-balancing cluster, use the **cluster key** command in VPN load-balancing mode. To remove this specification, use the **no** form of this command.

cluster key *shared-secret*

no cluster key [*shared-secret*]

Syntax Description

shared-secret	A string defining the shared secret for the VPN load-balancing cluster.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

You must first use the **vpn load-balancing** command to enter VPN load-balancing mode. The secret defined in the **cluster key** command is also used for cluster encryption.

You must use the **cluster key** command to configure the shared secret before enabling cluster encryption.

If you specify a value for *shared-secret* in the **no cluster key** form of the command, the shared secret value must match the existing configuration.

Examples

The following is an example of a VPN load-balancing command sequence that includes a **cluster key** command that sets the shared secret of the virtual load-balancing cluster to 123456789:

```
hostname(config)# interface GigabitEthernet 0/1
hostname(config-if)# ip address 209.165.202.159 255.255.255.0
hostname(config)# nameif test
hostname(config)# interface GigabitEthernet 0/2
hostname(config-if)# ip address 209.165.201.30 255.255.255.0
hostname(config)# nameif foo
hostname(config)# vpn load-balancing
hostname(config-load-balancing)# interface lbpublic test
hostname(config-load-balancing)# interface lbprivate foo
hostname(config-load-balancing)# cluster ip address 209.165.202.224
```

```
hostname(config-load-balancing)# cluster key 123456789
hostname(config-load-balancing)# cluster encryption
hostname(config-load-balancing)# participate
```

Related Commands

Command	Description
vpn load-balancing	Enters VPN load-balancing mode.

cluster port

To set the UDP port for the virtual load-balancing cluster, use the **cluster port** command in VPN load-balancing mode. To remove the port specification, use the **no** form of this command.

cluster port *port*

no cluster port [*port*]

Syntax Description

port	The UDP port that you want to assign to the virtual load-balancing cluster.

Defaults

The default cluster port is 9023.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

You must first use the **vpn load-balancing** command to enter VPN load-balancing mode.

You can specify any valid UDP port number. The range is 1-65535.

If you specify a value for *port* in the **no cluster port** form of the command, the port number specified must match the existing configured port number.

Examples

The following is an example of a VPN load-balancing command sequence that includes a **cluster port address** command that sets the UDP port for the virtual load-balancing cluster to 9023:

```
hostname(config)# interface GigabitEthernet 0/1
hostname(config-if)# ip address 209.165.202.159 255.255.255.0
hostname(config)# nameif test
hostname(config)# interface GigabitEthernet 0/2
hostname(config-if)# ip address 209.165.201.30 255.255.255.0
hostname(config)# nameif foo
hostname(config)# vpn load-balancing
hostname(config-load-balancing)# interface lbpublic test
hostname(config-load-balancing)# interface lbprivate foo
hostname(config-load-balancing)# cluster ip address 209.165.202.224
hostname(config-load-balancing)# cluster port 9023
hostname(config-load-balancing)# participate
```

Related Commands

Command	Description
vpn load-balancing	Enters VPN load-balancing mode.

command-alias

To create an alias for a command, use the **command-alias** command in global configuration mode. To remove the alias, use the **no** form of this command. When you enter the command alias, the original command is invoked. You might want to create command aliases to provide shortcuts for long commands, for example.

command-alias *mode command_alias original_command*

no command-alias *mode command_alias original_command*

Syntax Description

mode	Specifies the command mode in which you want to create the command alias, for example **exec** (for user and privileged EXEC modes), **configure**, or **interface**.
command_alias	Specifies the new name you want for an existing command.
original_command	Specifies the existing command or command with its keywords for which you want to create the command alias.

Defaults

By default, the following user EXEC mode aliases are configured:

h for **help**

lo for **logout**

p for **ping**

s for **show**

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

You can create an alias for the first part of any command and still enter the additional keywords and arguments as normal.

When you use CLI help, command aliases are indicated by an asterisk (*), and displayed in the following format:

```
*command-alias=original-command
```

For example, the **lo** command alias displays along with other privileged EXEC mode commands that start with "lo," as follows:

```
hostname# lo?
*lo=logout login  logout
```

You can use the same alias in different modes. For example, you can use "happy" in privileged EXEC mode and configuration mode to alias different commands, as follows:

```
hostname(config)# happy?

configure mode commands/options:
*happy="username crichton password test"

exec mode commands/options:
*happy=enable
```

To list only commands and omit aliases, begin your input line with a space. Also, to circumvent command aliases, use a space before entering the command. In the following example, the alias happy is not shown, because there is a space before the happy? command.

```
hostname(config)# alias exec test enable
hostname(config)# exit
hostname#   happy?
ERROR: % Unrecognized command
```

As with commands, you can use CLI help to display the arguments and keywords that can follow a command alias.

You must enter the complete command alias. Shortened aliases are not accepted. In the following example, the parser does not recognize the command hap as indicating the alias happy:

```
hostname# hap
% Ambiguous command: "hap"
```

Examples

The following example shows how to create a command alias named "**save**" for the **copy running-config startup-config** command:

```
hostname(config)# command-alias exec save copy running-config startup-config
hostname(config)# exit
hostname# save

Source filename [running-config]?
Cryptochecksum: 50d131d9 8626c515 0c698f7f 613ae54e

2209 bytes copied in 0.210 secs
hostname#
```

Related Commands

Command	Description
clear configure command-alias	Clears all non-default command aliases.
show running-config command-alias	Displays all non-default command aliases configured.

command-queue

To specify the maximum number of MGCP commands that are queued while waiting for a response, use the **command-queue** command in MGCP map configuration mode. To remove the configuration, use the **no** form of this command.

command-queue *limit*

no command-queue *limit*

Syntax Description

limit	Specifies the maximum number of commands to queue, from 1 to 2147483647.

Defaults

This command is disabled by default.

The default for the MGCP command queue is 200.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Use the **command-queue** command to specify the maximum number of MGCP commands that are queued while waiting for a response. The range of allowed values is from 1 to 4294967295. The default is 200. When the limit has been reached and a new command arrives, the command that has been in the queue for the longest time is removed.

Examples

The following example limits the MGCP command queue to 150 commands:

```
hostname(config)# mgcp-map mgcp_policy
hostname(config-mgcp-map)#command-queue 150
```

Related Commands

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

Commands	Description
show mgcp	Displays MGCP configuration and session information.
timeout	Configures the idle timeout after which an MGCP media or MGCP PAT xlate connection will be closed.

compatible rfc1583

To restore the method that is used to calculate the summary route costs per RFC 1583, use the **compatible rfc1583** command in router configuration mode. To disable RFC 1583 compatibility, use the **no** form of this command.

compatible rfc1583

no compatible rfc1583

Syntax Description

This command has no arguments or keywords.

Defaults

This command is enabled by default.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Only the **no** form of this command appears in the configuration.

Examples

The following example shows how to disable RFC 1583-compatible route summary cost calculation:

```
hostname(config-router)# no compatible rfc1583
hostname(config-router)#
```

Related Commands

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

compression

To enable compression for SVC connections and WebVPN connections, use the **compression** command from global configuration mode:

compression {**all** | **svc** | **http-comp**}

[**no**] **compression** {**all** | **svc** | **http-comp**}

To remove the command from the configuration, use the **no** form of the command.

Syntax Description

all	Specifies enabling all available compression techniques.
svc	Specifies compression for SVC connections.
http-comp	Specifies compression for WebVPN connections.

Defaults

The default is *all*. All available compression techniques are enabled.

Command Modes

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

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

Command History

Release	Modification
7.1.1	This command was introduced.

Usage Guidelines

For SVC connections, the **compression** command configured from global configuration mode overrides the **svc compression** command configured in group policy webvpn and username webvpn modes.

For example, if you enter the **svc compression** command for a certain group from group policy webvpn mode, and then you enter **no compression** command from global configuration mode, you override the **svc compression** command settings that you configured for the group.

Conversely, if you turn compression back on with the **compression** command from global configuration mode, any group settings take effect, and those settings ultimately determine the compression behavior.

If you disable compression with the **no compression** command, only new connections are affected. Active connections remain unaffected.

Examples

In the following example, compression is turned on for SVC connections:

```
hostname(config)# compression svc
```

In the next example, compression is disabled for SVC and WebVPN connections:

```
hostname(config)# no compression svc http-comp
```

Related Commands

Command	Description
show webvpn svc	Displays information about the SVC installation.
svc	Enables or requires the SVC for a specific group or user.
svc compression	Enables compression of http data over an SVC connection for a specific group or user.

config-register

To set the configuration register value that is used the next time you reload the security appliance, use the **config-register** command in global configuration mode. To set the value back to the default, use the **no** form of this command. This command is only supported on the ASA 5500 adaptive security appliance. The configuration register value determines which image to boot from as well as other boot parameters.

config-register *hex_value*

no config-register

Syntax Description

hex_value	Sets the configuration register value as a hexadecimal number from 0x0 to 0xFFFFFFFF. This number represents 32 bits and each hexadecimal character represents 4 bits. Each bit controls a different characteristic. However, bits 32 through 20 are either reserved for future use, cannot be set by the user, or are not currently used by the security appliance; therefore, you can ignore the three characters that represent those bits, because they are always set to 0. The relevent bits are represented by 5 hexadecimal characters: 0x*nnnnn*. You do not need to include preceding 0s. You do need to include trailing 0s. For example, 0x2001 is equivalent to 0x02001; but 0x10000 requires all the zeros. See Table 8-1 for more information about available values for the relevant bits.

Defaults

The default value is 0x1, which boots from the local image and startup configuration.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

The five characters are numbered from 0 to 4 from right to left, which is standard for hexadecimal and binary numbers. You can select one value for each character, and mix and match values as appropriate. For example, you can select either 0 or 2 for character number 3. Some values take priority if they conflict with other values. For example, if you set 0x2011, which sets the security appliance to both boot

from the TFTP server and to boot from the local image, the security appliance boots from the TFTP server. Because this value also stipulates that if the TFTP boot fails, the security appliance should boot directly into ROMMON, then the action that specifies to boot from the default image is ignored.

A value of 0 means no action unless otherwise specified.

Table 8-1 lists the actions associated with each hexadecimal character; choose one value for each character:

Table 8-1 Configuration Register Values

Prefix	Hexadecimal Character Numbers 4, 3, 2, 1, and 0				
0x	**0**	**0**	**0**	**0**	**0**
	1 Disables the 10 second ROMMON countdown during startup. Normally, you can press Escape during the countdown to enter ROMMON.	**2** If you set the security appliance to boot from a TFTP server, and the boot fails, then this value boots directly into ROMMON.	Reserved for future use.	**1** Boots from the TFTP server image as specified in the ROMMON Boot Parameters (which is the same as the **boot system tftp** command, if present). This value takes precedence over a value set for character 1. **Note** If character numbers 0 and 1 are not set to automatically boot an image, then the security appliance boots directly into ROMMON.	**1** Boots the image specified by the first **boot system** *local_flash* command. If that image does not load, the security appliance tries to boot each image specified by subsequent **boot system** commands until it boots successfully. **Note** If you disable password recovery using the **service password-recovery** command, then you cannot set the configuration register to ignore the startup configuration.
					3, **5**, **7**, **9** Boots the image specified by a particular **boot system** *local_flash* command. Value 3 boots the image specified in the first **boot system** command, value 5 boots the second image, and so on. If the image does not boot successfully, the security appliance does not attempt to fall back to other **boot system** command images (this is the difference between using value 1 and value 3). However, the security appliance has a failsafe feature that in the event of a boot failure attempts to boot from any image found in the root directory of internal Flash memory. If you do not want the failsafe feature to take effect, store your images in a different directory than root.
				4 Ignores the startup configuration and loads the default configuration.	**2**, **4**, **6**, **8** From ROMMON, if you enter the **boot** command without any arguments, then the security appliance boots the image specified by a particular **boot system** *local_flash* command. Value 3 boots the image specified in the first **boot system** command, value 5 boots the second image, and so on. This value does not automatically boot an image.
				5 Performs both actions above.	

The configuration register value is not replicated to a standby unit, but the following warning is displayed when you set the configuration register on the active unit:

```
WARNING The configuration register is not synchronized with the standby, their values may
not match.
```

You can also set the configuration register value in ROMMON using the **confreg** command.

Examples

The following example sets the configuration register to boot from the default image:

```
hostname(config)# config-register 0x1
```

Related Commands

Command	Description
boot	Sets the boot image and startup configuration.
service password-recovery	Enables or disables password recovery.

configure factory-default

To restore the configuration to the factory default, use the **configure factory-default** command in global configuration mode. The factory default configuration is the configuration applied by Cisco to new security appliances. This command is not supported on all platforms; see the CLI help for the **configure** command to confirm if the command is supported (enter **configure ?** at the global configuration prompt). The factory default configuration automatically configures an interface for management so you can connect to it using ASDM, with which you can then complete your configuration.

configure factory-default [*ip_address* [*mask*]]

Syntax Description

ip_address	Sets the IP address of the management interface, instead of using the default address, 192.168.1.1. If your platform includes a dedicated management interface, then this IP address applies to that interface. If your platform includes only data interfaces, then this address applies to the Ethernet 1 interface.
mask	Sets the subnet mask of the interface. If you do not set a mask, the security appliance uses the mask appropriate for the IP address class.

Defaults

The default IP address and mask are 192.168.1.1 and 255.255.255.0.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

The **configure factory-default** command configures the minimum commands required to connect to the security appliance using ASDM. This command is available only for routed firewall mode; transparent mode does not support IP addresses for interfaces, and setting the interface IP address is one of the actions this command takes. This command is also only available in single context mode; a security appliance with a cleared configuration does not have any defined contexts to automatically configure using this command.

This command clears the current running configuration and then configures several commands. The configured interface depends on your platform. For a platform with a dedicated management interface, the interface is named “management.” For other platforms, the configured interface is Ethernet 1 and named “inside.”

The following commands apply to the dedicated management interface, Management 0/0 (for a platform without a dedicated management interface, the interface is Ethernet 1):

```
interface management 0/0
   ip address 192.168.1.1 255.255.255.0
   nameif management
   security-level 100
   no shutdown
asdm logging informational 100
asdm history enable
http server enable
http 192.168.1.0 255.255.255.0 management
dhcpd address 192.168.1.2-192.168.1.254 management
dhcpd lease 3600
dhcpd ping_timeout 750
dhcpd enable management
```

If you set the IP address in the **configure factory-default** command, then the **http** command uses the subnet you specify. Similarly, the **dhcpd address** command range consists of addresses within the subnet that you specify.

After you restore the factory default configuration, save it to internal Flash memory using the **copy running-config startup-config** command. The **copy** command saves the running configuration to the default location for the startup configuration, even if you previously configured the **boot config** command to set a different location; when the configuration was cleared, this path was also cleared.

Note This command also clears the **boot system** command, if present, along with the rest of the configuration. The **boot system** command lets you boot from a specific image, including an image on the external Flash memory card. The next time you reload the security appliance after restoring the factory configuration, it boots from the first image in internal Flash memory; if you do not have an image in internal Flash memory, the security appliance does not boot.

To configure additional settings that are useful for a full configuration, see the **setup** command.

Examples

The following example resets the configuration to the factory default, assigns the IP address 10.1.1.1 to the interface, and then saves the new configuration as the startup configuration:

```
hostname(config)# configure factory-default 10.1.1.1 255.255.255.0
Based on the inside IP address and mask, the DHCP address
pool size is reduced to 253 from the platform limit 256

WARNING: The boot system configuration will be cleared.
The first image found in disk0:/ will be used to boot the
system on the next reload.
Verify there is a valid image on disk0:/ or the system will
not boot.

Begin to apply factory-default configuration:
Clear all configuration
...
hostname(config)#
hostname(config)# copy running-config startup-config
```

Related Commands

Command	Description
boot system	Sets the software image from which to boot.
clear configure	Clears the running configuration.
copy running-config startup-config	Copies the running configuration to the startup configuration.
setup	Prompts you to configure basic settings for the security appliance.
show running-config	Shows the running configuration.

configure http

To merge a configuration file from an HTTP(S) server with the running configuration, use the **configure http** command in global configuration mode. This command supports IPv4 and IPv6 addresses.

configure http[**s**]**://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path/*]*filename*

Syntax Description

:password	(Optional) For HTTP(S) authentication, specifies the password.
:*port*	(Optional) Specifies the port. For HTTP, the default is 80. For HTTPS, the default is 443.
@	(Optional) If you enter a name and/or a password, precedes the server IP address with an at sign (@).
filename	Specifies the configuration filename.
http[**s**]	Specifies either HTTP or HTTPS.
path	(Optional) Specifies a path to the filename.
server	Specifies the server IP address or name. For IPv6 server addresses, if you specify the port, then you must enclose the IP address in brackets so that the colons in the IP address are not mistaken for the colon before the port number. For example, enter the following address and port: `[fe80::2e0:b6ff:fe01:3b7a]:8080`
user	(Optional) For HTTP(S) authentication, specifies the username.

Defaults

For HTTP, the default port is 80. For HTTPS, the default port is 443.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

A merge adds all commands from the new configuration to the running configuration, and overwrites any conflicting commands with the new versions. For example, if a command allows multiple instances, the new commands are added to the existing commands in the running configuration. If a command allows only one instance, the new command overwrites the command in the running configuration. A merge never removes commands that exist in the running configuration but are not set in the new configuration.

This command is the same as the **copy http running-config** command. For multiple context mode, that command is only available in the system execution space, so the **configure http** command is an alternative for use within a context.

Examples

The following example copies a configuration file from an HTTPS server to the running configuration:

```
hostname(config)# configure https://user1:pa$$w0rd@10.1.1.1/configs/newconfig.cfg
```

Related Commands

Command	Description
clear configure	Clears the running configuration.
configure memory	Merges the startup configuration with the running configuration.
configure net	Merges a configuration file from the specified TFTP URL with the running configuration.
configure factory-default	Adds commands you enter at the CLI to the running configuration.
show running-config	Shows the running configuration.

configure memory

To merge the startup configuration with the running configuration, use the **configure memory** command in global configuration mode.

configure memory

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

A merge adds all commands from the new configuration to the running configuration, and overwrites any conflicting commands with the new versions. For example, if a command allows multiple instances, the new commands are added to the existing commands in the running configuration. If a command allows only one instance, the new command overwrites the command in the running configuration. A merge never removes commands that exist in the running configuration but are not set in the new configuration.

If you do not want to merge the configurations, you can clear the running configuration, which disrupts any communications through the security appliance, and then enter the **configure memory** command to load the new configuration.

This command is equivalent to the **copy startup-config running-config** command.

For multiple context mode, a context startup configuration is at the location specified by the **config-url** command.

Examples

The following example copies the startup configuration to the running configuration:

```
hostname(config)# configure memory
```

Related Commands

Command	Description
clear configure	Clears the running configuration.
configure http	Merges a configuration file from the specified HTTP(S) URL with the running configuration.
configure net	Merges a configuration file from the specified TFTP URL with the running configuration.
configure factory-default	Adds commands you enter at the CLI to the running configuration.
show running-config	Shows the running configuration.

configure net

To merge a configuration file from a TFTP server with the running configuration, use the **configure net** command in global configuration mode. This command supports IPv4 and IPv6 addresses.

configure net [*server*:[*filename*] | :*filename*]

Syntax Description

:*filename*	Specifies the path and filename. If you already set the filename using the **tftp-server** command, then this argument is optional. If you specify the filename in this command as well as a name in the **tftp-server** command, the security appliance treats the **tftp-server** command filename as a directory, and adds the **configure net** command filename as a file under the directory. To override the **tftp-server** command value, enter a slash in front of the path and filename. The slash indicates that the path is not relative to the tftpboot directory, but is an absolute path. The URL generated for this file includes a double slash (//) in front of the filename path. If the file you want is in the tftpboot directory, you can include the path for the tftpboot directory in the filename path. If you specified the TFTP server address using the **tftp-server** command, you can enter the filename alone preceded by a colon (:).
server:	Sets the TFTP server IP address or name. This address overrides the address you set in the **tftp-server** command, if present. For IPv6 server addresses, you must enclose the IP address in brackets so that the colons in the IP address are not mistaken for the colon before the filename. For example, enter the following address: [fe80::2e0:b6ff:fe01:3b7a] The default gateway interface is the highest security interface; however, you can set a different interface name using the **tftp-server** command.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

A merge adds all commands from the new configuration to the running configuration, and overwrites any conflicting commands with the new versions. For example, if a command allows multiple instances, the new commands are added to the existing commands in the running configuration. If a command allows only one instance, the new command overwrites the command in the running configuration. A merge never removes commands that exist in the running configuration but are not set in the new configuration.

This command is the same as the **copy tftp running-config** command. For multiple context mode, that command is only available in the system execution space, so the **configure net** command is an alternative for use within a context.

Examples

The following example sets the server and filename in the **tftp-server** command, and then overrides the server using the **configure net** command. The same filename is used.

```
hostname(config)# tftp-server inside 10.1.1.1 configs/config1
hostname(config)# configure net 10.2.2.2:
```

The following example overrides the server and the filename. The default path to the filename is /tftpboot/configs/config1. The /tftpboot/ part of the path is included by default when you do not lead the filename with a slash (/). Because you want to override this path, and the file is also in tftpboot, include the tftpboot path in the **configure net** command.

```
hostname(config)# tftp-server inside 10.1.1.1 configs/config1
hostname(config)# configure net 10.2.2.2:/tftpboot/oldconfigs/config1
```

The following example sets the server only in the **tftp-server** command. The **configure net** command specifies only the filename.

```
hostname(config)# tftp-server inside 10.1.1.1
hostname(config)# configure net :configs/config1
```

Related Commands

Command	Description
configure http	Merges a configuration file from the specified HTTP(S) URL with the running configuration.
configure memory	Merges the startup configuration with the running configuration.
show running-config	Shows the running configuration.
tftp-server	Sets a default TFTP server and path for use in other commands.
write net	Copies the running configuration to a TFTP server.

configure terminal

To configure the running configuration at the command line, use the **configure terminal** command in privileged EXEC mode. This command enters global configuration mode, which lets you enter commands that change the configuration.

configure terminal

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Examples

The following example enters global configuration mode:

```
hostname# configure terminal
hostname(config)#
```

Related Commands

Command	Description
clear configure	Clears the running configuration.
configure http	Merges a configuration file from the specified HTTP(S) URL with the running configuration.
configure memory	Merges the startup configuration with the running configuration.
configure net	Merges a configuration file from the specified TFTP URL with the running configuration.
show running-config	Shows the running configuration.

config-url

To identify the URL from which the system downloads the context configuration, use the **config-url** command in context configuration mode.

config-url *url*

Syntax Description

url	Sets the context configuration URL. All remote URLs must be accessible from the admin context. See the following URL syntax: • **disk0:/**[*path*/]*filename* For the ASA 5500 series adaptive security appliance, this URL indicates the internal Flash memory. You can also use **flash** instead of **disk0**; they are aliased. • **disk1:/**[*path*/]*filename* For the ASA 5500 series adaptive security appliance, this URL indicates the external Flash memory card. • **flash:/**[*path*/]*filename* This URL indicates the internal Flash memory. • **ftp://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path*/]*filename*[**;type=***xx*] The **type** can be one of the following keywords: – **ap**—ASCII passive mode – **an**—ASCII normal mode – **ip**—(Default) Binary passive mode – **in**—Binary normal mode • **http**[**s**]**://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path*/]*filename* • **tftp://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path*/]*filename*[**;int=***interface_name*] Specify the interface name if you want to override the route to the server address.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

When you add a context URL, the system immediately loads the context so that it is running.

Note Enter the **allocate-interface** command(s) before you enter the **config-url** command. The security appliance must assign interfaces to the context before it loads the context configuration; the context configuration might include commands that refer to interfaces (**interface**, **nat**, **global**...). If you enter the **config-url** command first, the security appliance loads the context configuration immediately. If the context contains any commands that refer to interfaces, those commands fail.

The filename does not require a file extension, although we recommend using ".cfg".

The admin context file must be stored on the internal Flash memory.

If you download a context configuration from an HTTP or HTTPS server, you cannot save changes back to these servers using the **copy running-config startup-config** command. You can, however, use the **copy tftp** command to copy the running configuration to a TFTP server.

If the system cannot retrieve the context configuration file because the server is unavailable, or the file does not yet exist, the system creates a blank context that is ready for you to configure with the command-line interface.

To change the URL, reenter the **config-url** command with a new URL.

The security appliance merges the new configuration with the current running configuration. Reentering the same URL also merges the saved configuration with the running configuration. A merge adds any new commands from the new configuration to the running configuration. If the configurations are the same, no changes occur. If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results. If the running configuration is blank (for example, if the server was unavailable and the configuration was never downloaded), then the new configuration is used. If you do not want to merge the configurations, you can clear the running configuration, which disrupts any communications through the context, and then reload the configuration from the new URL.

Examples

The following example sets the admin context to be "administrator," creates a context called "administrator" on the internal Flash memory, and then adds two contexts from an FTP server:

```
hostname(config)# admin-context administrator
hostname(config)# context administrator
hostname(config-ctx)# allocate-interface gigabitethernet0/0.1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.1
hostname(config-ctx)# config-url flash:/admin.cfg

hostname(config-ctx)# context test
hostname(config-ctx)# allocate-interface gigabitethernet0/0.100 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/0.102 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115
int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg

hostname(config-ctx)# context sample
hostname(config-ctx)# allocate-interface gigabitethernet0/1.200 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.212 int2
```

```
hostname(config-ctx)# allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235
int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
```

Related Commands

Command	Description
allocate-interface	Allocates interfaces to a 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.

console timeout

To set the idle timeout for a console connection to the security appliance, use the **console timeout** command in global configuration mode. To disable, use the **no** form of this command.

console timeout *number*

no console timeout [*number*]

Syntax Description

number	Specifies the idle time in minutes (0 through 60) after which the console session ends.

Defaults

The default timeout is 0, which means the console session will not time out.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

The **console timeout** command sets the timeout value for any authenticated, enable mode, or configuration mode user session to the security appliance. The **console timeout** command does not alter the Telnet or SSH timeouts; these access methods maintain their own timeout values.

The **no console timeout** command resets the console timeout value to the default timeout of 0, which means that the console will not time out.

Examples

The following example shows how to set the console timeout to 15 minutes:

```
hostname(config)# console timeout 15
```

Related Commands

Command	Description
clear configure console	Restores the default console connection settings.
clear configure timeout	Restores the default idle time durations in the configuration.
show running-config console timeout	Displays the idle timeout for a console connection to the security appliance.

content-length

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

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

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

Syntax Description

action	Specifies the action taken when a message fails this inspection.
allow	Allows the message.
bytes	Specifies the number of bytes. The permitted range is 1 to 65535 for the **min** option and 1 to 50000000 for the **max** option.
drop	Closes the connection.
log	(Optional) Generates a syslog.
max	(Optional) Specifies the maximum content length allowed.
min	Specifies the minimum content length allowed.
reset	Sends a TCP reset message to client and server.

Defaults

This command is disabled by default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

After enabling the **content-length** command, the security appliance only allows messages within the configured range and otherwise takes the specified action. Use the **action** keyword to cause the security appliance to reset the TCP connection and create a syslog entry.

Examples

The following example restricts HTTP traffic to messages 100 bytes or larger and not exceeding 2000 bytes. If a message is outside this range, the security appliance resets the TCP connection and creates a syslog entry.

```
hostname(config)# http-map inbound_http
hostname(config-http-map)# content-length min 100 max 2000 action reset log
hostname(config-http-map)# exit
```

Related Commands

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

content-type-verification

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

content-type-verification [**match-req-rsp**] **action** {**allow** | **reset** | **drop**} [**log**]

no content-type-verification [**match-req-rsp**] **action** {**allow** | **reset** | **drop**} [**log**]

Syntax Description

action	Specifies the action taken when a message fails command inspection.
allow	Allows the message.
drop	Closes the connection.
log	(Optional) Generates a syslog message.
match-req-rsp	(Optional) Verifies that the content-type field in the HTTP response matches the accept field in the corresponding HTTP request message.
reset	Sends a TCP reset message to client and server.

Defaults

This command is disabled by default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced

Usage Guidelines

This command enables the following checks:

- Verifies that the value of the header content-type is in the internal list of supported content types,
- Verifies that the header content-type matches the actual content in the data or entity body portion of the message.
- The **match-req-rsp** keyword enables an additional check that verifies the content-type field in the HTTP response matches the **accept** field in the corresponding HTTP request message.

If the message fails any of the above checks, the security appliance takes the configured action.

The following is the list of supported content types.

audio/* |	audio/basic |	video/x-msvideo
audio/mpeg |	audio/x-adpcm |	audio/midi
audio/x-ogg |	audio/x-wav |	audio/x-aiff |
application/octet-stream	application/pdf	application/msword
application/vnd.ms-excel	application/vnd.ms-powerpoint	application/postscript
application/x-java-arching	application/x-msn-messenger	application/x-gzip
image |	application/x-java-xm	application/zip
image/jpeg |	image/cgf |	image/gif |
image/x-3ds |	image/png |	image/tiff |
image/x-portable-bitmap |	image/x-bitmap |	image/x-niff |
text/* |	image/x-portable-greymap |	image/x-xpm |
text/plain |	text/css	text/html |
text/xmcd	text/richtext |	text/sgml
video/-flc	text/xml	video/*
video/sgi	video/mpeg	video/quicktime
video/x-mng	video/x-avi	video/x-fli

Some content-types in this list may not have a corresponding regular expression (magic number) so they cannot be verified in the body portion of the message. When this case occurs, the HTTP message will be allowed.

Examples

The following example restricts HTTP traffic based on the content type of the HTTP message. If a message contains an unsupported content type, the security appliance resets the TCP connection and creates a syslog entry.

```
hostname(config)# http-map inbound_http
hostname(config-http-map)# content-type-verification match-req-rsp reset log
hostname(config-http-map)# exit
```

Related Commands

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

context

To create a security context in the system configuration and enter context configuration mode, use the **context** command in global configuration mode. To remove a context, use the **no** form of this command. In context configuration mode, you can identify the configuration file URL and interfaces that a context can use.

context *name*

no context *name* [**noconfirm**]

Syntax Description

name	Sets the name as a string up to 32 characters long. This name is case sensitive, so you can have two contexts named "customerA" and "CustomerA," for example. You can use letters, digits, or hyphens, but you cannot start or end the name with a hyphen. "System" or "Null" (in upper or lower case letters) are reserved names, and cannot be used.
noconfirm	(Optional) Removes the context without prompting you for confirmation. This option is useful for automated scripts.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

If you do not have an admin context (for example, if you clear the configuration) then the first context you add must be the admin context. To add an admin context, see the **admin-context** command. After you specify the admin context, you can enter the **context** command to configure the admin context.

You can only remove a context by editing the system configuration. You cannot remove the current admin context using the **no** form of this command; you can only remove it if you remove all contexts using the **clear configure context** command.

Examples

The following example sets the admin context to be "administrator," creates a context called "administrator" on the internal Flash memory, and then adds two contexts from an FTP server:

```
hostname(config)# admin-context administrator
hostname(config)# context administrator
hostname(config-ctx)# allocate-interface gigabitethernet0/0.1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.1
hostname(config-ctx)# config-url flash:/admin.cfg

hostname(config-ctx)# context test
hostname(config-ctx)# allocate-interface gigabitethernet0/0.100 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/0.102 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115
int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg

hostname(config-ctx)# context sample
hostname(config-ctx)# allocate-interface gigabitethernet0/1.200 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.212 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235
int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
```

Related Commands

Command	Description
allocate-interface	Assigns interfaces to a context.
changeto	Changes between contexts and the system execution space.
config-url	Specifies the location of the context configuration.
join-failover-group	Assigns a context to a failover group.
show context	Shows context information.

copy

To copy a file from one location to another, use the **copy** command.

copy [**/noconfirm** | **/pcap**] {*url* | **running-config** | **startup-config**}
{**running-config** | **startup-config** | *url*}

Syntax Description

/noconfirm	Copies the file without a confirmation prompt.
/pcap	Specifies the defaults of the preconfigured TFTP server. See the **tftp-server** command to configure a default TFTP server.
running-config	Specifies the running configuration.

startup-config	Specifies the startup configuration. The startup configuration for single mode or for the system in multiple context mode is a hidden file in Flash memory. From within a context, the location of the startup configuration is specified by the **config-url** command. For example, if you specify an HTTP server for the **config-url** command and then enter the **copy startup-config running-config** command, the security appliance copies the startup configuration from the HTTP server using the admin context interface.
url	Specifies the source or destination file to be copied. Not all combinations of source and destination URLs are allowed. For example, you cannot copy from a remote server to another remote server; this command is meant to copy between local and remote locations. In a context, you can copy the running or startup configuration to a TFTP or FTP server using the context interfaces, but you cannot copy from a server to the running or startup configuration. See the **startup-config** keyword for other options. Also, see the **configure net** command to download from a TFTP server to the running context configuration. See the following URL syntax: • **disk0:/**[*path*/]*filename* This option is only available for the ASA 5500 series adaptive security appliance, and indicates the internal Flash memory. You can also use **flash** instead of **disk0**; they are aliased. • **disk1:/**[*path*/]*filename* This option is only available for the ASA 5500 series adaptive security appliance, and indicates the external Flash memory card. • **flash:/**[*path*/]*filename* This option indicates the internal Flash card. For the ASA 5500 series adaptive security appliance, **flash** is an alias for **disk0**. • **ftp://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path*/]*filename*[**;type=***xx*] The **type** can be one of the following keywords: – **ap**—ASCII passive mode – **an**—ASCII normal mode – **ip**—(Default) Binary passive mode – **in**—Binary normal mode • **http**[**s**]**://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path*/]*filename* • **tftp://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path*/]*filename*[**;int=***interface_name*] Specify the interface name if you want to override the route to the server address. The pathname cannot contain spaces. If a pathname has spaces, set the path in the **tftp-server** command instead of in the **copy tftp** command.

Defaults

This command has no default settings.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

When you copy a configuration to the running configuration, you merge the two configurations. A merge adds any new commands from the new configuration to the running configuration. If the configurations are the same, no changes occur. If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results.

Examples

This example shows how to copy a file from the disk to a TFTP server in the system execution space:

```
hostname(config)# copy disk0:my_context/my_context.cfg
tftp://10.7.0.80/my_context/my_context.cfg
```

This example shows how to copy a file from one location on the disk to another location on the disk. The name of the destination file can be either the name of the source file or a different name.

```
hostname(config)# copy disk0:my_context.cfg disk:my_context/my_context.cfg
```

This example shows how to copy an ASDM file from a TFTP server to the internal Flash memory:

```
hostname(config)# copy tftp://10.7.0.80/asdm700.bin disk0:asdm700.bin
```

This example shows how to copy the running configuration in a context to a TFTP server:

```
hostname(config)# copy running-config tftp://10.7.0.80/my_context/my_context.cfg
```

Related Commands

Command	Description
configure net	Copies a file from a TFTP server to the running configuration.
copy capture	Copies a capture file to a TFTP server.
tftp-server	Sets the default TFTP server.
write memory	Saves the running configuration to the startup configuration.
write net	Copies the running configuration to a TFTP server.

copy capture

To copy a capture file to a server, use the **copy capture** command in privileged EXEC mode.

copy [**/noconfirm**] [**/pcap**] **capture:** [*context_name***/**]*buffer_name url*

Syntax Description

/noconfirm	Copies the file without a confirmation prompt.
/pcap	Copies the packet capture as raw data.
buffer_name	Unique name that identifies the capture.
*context_name***/**	Copies a packet capture defined in a security context.
url	Specifies the destination to copy the packet capture file. See the following URL syntax: • **disk0:/**[*path***/**]*filename* This option is only available for the ASA 5500 series adaptive security appliance, and indicates the internal Flash card. You can also use **flash** instead of **disk0**; they are aliased. • **disk1:/**[*path***/**]*filename* This option is only available for the ASA 5500 series adaptive security appliance, and indicates the external Flash card. • **flash:/**[*path***/**]*filename* This option indicates the internal Flash card. For the ASA 5500 series adaptive security appliance, **flash** is an alias for **disk0**. • **ftp://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path***/**]*filename*[**;type=***xx*] The **type** can be one of the following keywords: – **ap**—ASCII passive mode – **an**—ASCII normal mode – **ip**—(Default) Binary passive mode – **in**—Binary normal mode • **http**[**s**]**://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path***/**]*filename* • **tftp://**[*user*[**:***password*]@]*server*[:*port*]**/**[*path***/**]*filename*[**;int=***interface_name*] Specify the interface name if you want to override the route to the server address. The pathname cannot contain spaces. If a pathname has spaces, set the path in the **tftp-server** command instead of in the **copy tftp** command.

Defaults

This command has no default settings.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Examples

The following example shows the prompts that are provided when you enter the **copy capture** command without specifying the full path:

```
hostname(config)# copy capture:abc tftp
Address or name of remote host [171.68.11.129]?
Source file name [username/cdisk]?
copying capture to tftp://171.68.11.129/username/cdisk:
[yes|no|again]? y
!!!!!!!!!!!!!
```

You can specify the full path as follows:

```
hostname(config)# copy capture:abc tftp:171.68.11.129/tftpboot/abc.cap
```

If the TFTP server is already configured, the location or filename can be unspecified as follows:

```
hostname(config)# tftp-server outside 171.68.11.129 tftp/cdisk
hostname(config)# copy capture:abc tftp:/tftp/abc.cap
```

Related Commands

Command	Description
capture	Enables packet capabilities for packet sniffing and network fault isolation.
clear capture	Clears the capture buffer.
show capture	Displays the capture configuration when no options are specified.

crashinfo console disable

To read, write, and configure crash write to flash, use the **crashinfo console disable** command.

crashinfo console disable

[no] crashinfo console disable

Syntax Description

disable	Suppresses console output in the event of a crash.

Defaults

This command has no default settings.

Command Modes

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

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

Command History

Release	Modification
7.0(4)	Support for this command was introduced.

Usage Guidelines

This command lets you suppress crashinfo from being output to the console. The crashinfo may contain sensitive information that is not appropriate for viewing by all users connected to the device. In conjunction with this command, you should also ensure crashinfo is written to flash, which can be examined after the device reboots. This command effects output for crashinfo and checkheaps, which is saved to flash and should be sufficient for troubleshooting.

Examples

```
hostname(config)# crashinfo console disable
```

Related Commands

Command	Description
clear configure fips	Clears the system or module FIPS configuration information stored in NVRAM.
fips enable	Enables or disablea policy-checking to enforce FIPS compliance on the system or module.
fips self-test poweron	Executes power-on self-tests.

Command	Description
show crashinfo console	Reads, writes, and configures crash write to flash.
show running-config fips	Displays the FIPS configuration that is running on the security appliance.

crashinfo force

To force the security appliance to crash, use the **crashinfo force** command in privileged EXEC mode.

crashinfo force [**page-fault | watchdog**]

Syntax Description

page-fault	(Optional) Forces a crash of the security appliance as a result of a page fault.
watchdog	(Optional) Forces a crash of the security appliance as a result of watchdogging.

Defaults

The security appliance saves the crash information file to Flash memory by default.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

You can use the **crashinfo force** command to test the crash output generation. In the crash output, there is nothing that differentiates a real crash from a crash resulting from the **crashinfo force page-fault** or **crashinfo force watchdog** command (because these are real crashes). The security appliance reloads after the crash dump is complete.

Caution Do not use the **crashinfo force** command in a production environment. The **crashinfo force** command crashes the security appliance and forces it to reload.

Examples

The following example shows the warning that displays when you enter the **crashinfo force page-fault** command:

```
hostname# crashinfo force page-fault
WARNING: This command will force the XXX to crash and reboot.
    Do you wish to proceed? [confirm]:
```

If you enter a carriage return (by pressing the Return or Enter key on your keyboard), "**Y**", or "**y**" the security appliance crashes and reloads; any of these responses are interpreted as confirmation. Any other character is interpreted as a **no**, and the security appliance returns to the command-line prompt.

Related Commands

clear crashinfo	Clears the contents of the crash information file.
crashinfo save disable	Disables crash information from writing to Flash memory.
crashinfo test	Tests the ability of the security appliance to save crash information to a file in Flash memory.
show crashinfo	Displays the contents of the crash information file.

crashinfo save disable

To disable crash information from writing to Flash memory, use the **crashinfo save** command in global configuration mode.

crashinfo save disable

no crashinfo save disable

Syntax Description

This command has no default arguments or keywords.

Defaults

The security appliance saves the crash information file to Flash memory by default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	The **crashinfo save enable** command was deprecated and is no longer a valid option. Use the **no crashinfo save disable** command instead.

Usage Guidelines

Crash information writes to Flash memory first, and then to your console.

Note If the security appliance crashes during startup, the crash information file is not saved. The security appliance must be fully initialized and running first, before it can save crash information to Flash memory.

Use the **no crashinfo save disable** command to re-enable saving the crash information to Flash memory.

Examples

```
hostname(config)# crashinfo save disable
```

Related Commands

clear crashinfo	Clears the contents of the crash file.
crashinfo force	Forces a crash of the security appliance.

crashinfo test	Tests the ability of the security appliance to save crash information to a file in Flash memory.
show crashinfo	Displays the contents of the crash file.

crashinfo test

To test the ability of the security appliance to save crash information to a file in Flash memory, use the **crashinfo test** command in global configuration mode.

crashinfo test

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

If a previous crash information file already exists in Flash memory, that file is overwritten.

Note Entering the **crashinfo test** command does not crash the security appliance.

Examples

The following example shows the output of a crash information file test.

```
hostname(config)# crashinfo test
```

Related Commands

clear crashinfo	Deletes the contents of the crash file.
crashinfo force	Forces the security appliance to crash.
crashinfo save disable	Disables crash information from writing to Flash memory.
show crashinfo	Displays the contents of the crash file.

crl

To specify CRL configuration options, use the **crl** command in crypto ca trustpoint configuration mode.

crl {**required** | **optional** | **nocheck**}

Syntax Description

required	The required CRL must be available for a peer certificate to be validated.
optional	The security appliance can still accept the peer certificate if the required CRL is not available.
nocheck	Directs the security appliance not to perform CRL checking.

Defaults

The default value is **nocheck**.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Crypto ca trustpoint configuration	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Examples

The following example enters crypto ca trustpoint configuration mode for trustpoint central, and requires that a CRL be available for a peer certificate to be validated for trustpoint central:

```
hostname(config)# crypto ca trustpoint central
hostname(ca-trustpoint)# crl required
hostname(ca-trustpoint)#
```

Related Commands

Command	Description
clear configure crypto ca trustpoint	Removes all trustpoints.
crypto ca trustpoint	Enters trustpoint submode.
crl configure	Enters crl configuration mode.

crl configure

To enter CRL configuration configuration mode, use the **crl configure** command in crypto ca trustpoint configuration mode.

crl configure

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Crypto ca trustpoint configuration	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Examples

The following example enters crl configuration mode within trustpoint central:

```
hostname<config># crypto ca trustpoint central
hostname<ca-trustpoint># crl configure
hostname<ca-crl>#
```

Related Commands

Command	Description
clear configure crypto ca trustpoint	Removes all trustpoints.
crypto ca trustpoint	Enters trustpoint submode.