so – st

software authenticity development

To enable or disable loading development key signed images, use the **software authenticity development** command in parameters configuration mode. Parameters configuration mode is accessible from policy map configuration mode. Once you enable this option, it persists until you disable loading development key signed images.

software authenticity development { **enable | disable** }

Syntax Description

disable	Disables loading development key signed images.
enable	Enables loading development key signed images.

Command Default

This command is disabled by default.

Command Modes

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

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

Command History

Release	Modification
9.3(2)	This command was added.

Examples

The following example shows loading development key signed signatures enabled:

```
ciscoasa(config)# software authenticity development enable
ciscoasa(config)# show software authenticity development
Loading of development images is enabled
ciscoasa(config)#
```

The following example shows loading development key signed images disabled:

```
ciscoasa(config)# software authenticity development disable
ciscoasa(config)# show software authenticity development
Loading of development images is disabled
ciscoasa(config)#
```

Related Commands

Command	Description
show software authenticity keys	Displays the development keys.

Command	Description
show software authenticity file disk0:asa932-1fbff.SSA	Displays the contents of the development key file.
show software authenticity running	Displays the digital signature information related to the current running file.
software authenticity key add special	Adds a new development key to SPI flash.
software authenticity key revoke special	Deletes older development keys from SPI flash.

software authenticity key add special

To add a new development key to the SPI flash, use the **software authenticity key add special** command in parameters configuration mode. Parameters configuration mode is accessible from policy map configuration mode.

software authenticity key add special

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
9.3(2)	This command was added.

Examples

The following example shows how to add a new development key to SPI flash:

```
ciscoasa(config)# software authenticity key add special
Writing the key to Primary...Success
Writing the key to Backup...Success
Done!
The following example shows what happens if you try to add a new development image to SPR
flash and one already exists:
ciscoasa(config)# software authenticity key add special
Duplicate key found in Primary...Skipping key write
Duplicate key found in Backup...Skipping key write
Done!
```

Related Commands

Command	Description
software authenticity key revoke special	Deletes older development keys from SPI flash.
show software authenticity keys	Displays the development keys in SPI flash.
show software authenticity file disk0:asa932-1fbff.SSA	Displays the contents of the development keys file.

Command	Description
show software authenticity running	Displays the digital signature information related to the current running file.

software authenticity key revoke special

To delete older development keys from SPI flash, use the **software authenticity key revoke special** command in parameters configuration mode. Parameters configuration mode is accessible from policy map configuration mode.

software authenticity key revoke special

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
9.3(2)	This command was added.

Examples

The following example shows how to remove a development key from SPI flash:

```
ciscoasa(config)# software authenticity key revoke special
Revoking the key with version A...Success
Revoking the key with version A...Success
Done!
```

Related Commands

Command	Description
software authenticity key add special	Adds a new development key to SPI flash.
show software authenticity keys	Displays the development keys in SPI flash.
show software authenticity file disk0:asa932-1fbff.SSA	Displays the contents of the development keys file.
show software authenticity running	Displays the digital signature information related to the current running file.

software-version

To identify the Server and User-Agent header fields, which expose the software version of either a server or an endpoint, use the **software-version** command in parameters configuration mode. Parameters configuration mode is accessible from policy map configuration mode. To disable this feature, use the **no** form of this command.

software-version action { **mask | log** } [**log**]

no software-version action { **mask | log** } [**log**]

Syntax Description

log	Specifies standalone or additional log in case of violation.
mask	Masks the software version in the SIP message.

Command Default

This command is disabled by default.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to identify the software version in a SIP inspection policy map:

```
ciscoasa(config)# policy-map type inspect sip sip_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# software-version action log
```

Related Commands

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

source-interface

To specify the source interface name for the VXLAN VTEP interface, use the **source-interface** command in nve configuration mode. To remove the interface, use the **no** form of this command.

source-interface*interface_name*
no source-interface *interface_name*

Syntax Description

interface_name	Sets the VTEP source interface name.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
9.4(1)	This command was added.

Usage Guidelines

The VTEP source interface is a regular ASA interface (physical, redundant, EtherChannel, or even VLAN) with which you plan to associate all VNI interfaces. You can configure one VTEP source interface per ASA/security context.

The VTEP source interface can be devoted wholly to VXLAN traffic, although it is not restricted to that use. If desired, you can use the interface for regular traffic and apply a security policy to the interface for that traffic. For VXLAN traffic, however, all security policy must be applied to the VNI interfaces. The VTEP interface serves as a physical port only.

In transparent firewall mode, the VTEP source interface is not part of a BVI, and you do configure an IP address for it, similar to the way the management interface is treated.

Note If the source interface MTU is less than 1554 bytes, then the ASA automatically raises the MTU to 1554 bytes.

Examples

The following example configures the GigabitEthernet 1/1 interface as the VTEP source interface:

```
ciscoasa(config)# interface gigabitethernet 1/1
ciscoasa(config-if)# nameif outside
```

```
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
ciscoasa(config-if)# nve 1
ciscoasa(cfg-nve)# source-interface outside
ciscoasa(cfg-nve)# default-mcast-group 236.0.0.100
```

Related Commands

Command	Description
debug vxlan	Debugs VXLAN traffic.
default-mcast-group	Specifies a default multicast group for all VNI interfaces associated with the VTEP source interface.
encapsulation vxlan	Sets the NVE instance to VXLAN encapsulation.
inspect vxlan	Enforces compliance with the standard VXLAN header format.
interface vni	Creates the VNI interface for VXLAN tagging.
mcast-group	Sets the multicast group address for the VNI interface.
nve	Specifies the Network Virtualization Endpoint instance.
nve-only	Specifies that the VXLAN source interface is NVE-only.
peer ip	Manually specifies the peer VTEP IP address.
segment-id	Specifies the VXLAN segment ID for a VNI interface.
show arp vtep-mapping	Displays MAC addresses cached on the VNI interface for IP addresses located in the remote segment domain and the remote VTEP IP addresses.
show interface vni	Shows the parameters, status and statistics of a VNI interface, status of its bridged interface (if configured), and NVE interface it is associated with.
show mac-address-table vtep-mapping	Displays the Layer 2 forwarding table (MAC address table) on the VNI interface with the remote VTEP IP addresses.
show nve	Shows the parameters, status and statistics of a NVE interface, status of its carrier interface (source interface), IP address of the carrier interface, VNIs that use this NVE as the VXLAN VTEP, and peer VTEP IP addresses associated with this NVE interface.
show vni vlan-mapping	Shows the mapping between VNI segment IDs and VLAN interfaces or physical interfaces in transparent mode.
source-interface	Specifies the VTEP source interface.
vtep-nve	Associates a VNI interface with the VTEP source interface.
vxlan port	Sets the VXLAN UDP port. By default, the VTEP source interface accepts VXLAN traffic to UDP port 4789.

speed

To set the speed of an interface, use the **speed** command in interface configuration mode. To restore the speed setting to the default, use the **no** form of this command.

speed { *speed* | **auto** | **nonegotiate** | **sfp-detect** }
no speed [*speed* | **auto** | **nonegotiate** | **sfp-detect**]

Syntax Description

auto	Auto detects the speed. RJ-45 only.
nonegotiate	For SFP interfaces (except for the Secure Firewall 3100), **no speed nonegotiate** sets the speed to 1000 Mbps and enables link negotiation for flow-control parameters and remote fault information. For 10 Gbps interfaces, this option sets the speed down to 1000 Mbps. The **nonegotiate** keyword is the only keyword available for SFP interfaces. The **speed nonegotiate** command disables link negotiation. For the Secure Firewall 3100, see the **negotiate-auto** command.
speed	Sets the speed to a specific setting.
sfp-detect	(Secure Firewall 3100 only) Detects the speed of the installed SFP module and uses the appropriate speed. Duplex is always full, and auto-negotiation is always enabled. This option is useful if you later change the network module to a different model, and want the speed to update automatically. This setting is the default.

Command Default

For RJ-45 interfaces, the default is **speed auto**.

For SFP interfaces (except for the Secure Firewall 3100), the default is **no speed nonegotiate**.

For the Secure Firewall 3100, the default is **sfp-detect**.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was moved from a keyword of the **interface** command to an interface configuration mode command.
9.14(1)	Speed auto-negotation can be disabled on 1GB fiber interfaces on the Firepower 1000 and 2100 using the **speed nonegotiate** command.
9.17(1)	We added the sfp-detect keyword for the Secure Firewall 3100.

Usage Guidelines

Set the speed on the physical interface only.

If your network does not support auto detection, set the speed to a specific value.

For RJ-45 interfaces on the ASA 5500 series, the default auto-negotiation setting also includes the Auto-MDI/MDIX feature. Auto-MDI/MDIX eliminates the need for crossover cabling by performing an internal crossover when a straight cable is detected during the auto-negotiation phase. Either the speed or duplex must be set to auto-negotiate to enable Auto-MDI/MDIX for the interface. If you explicitly set both the speed and duplex to a fixed value, thus disabling auto-negotiation for both settings, then Auto-MDI/MDIX is also disabled.

If you set the speed to anything other than **auto** on PoE ports, if available, then Cisco IP phones and Cisco wireless access points that do not support IEEE 802.3af will not be detected and supplied with power.

Note Do not set the **speed** command for an ASA 5500-X or an ASA 5585-X with fiber interfaces. Doing so causes a link failure.

Examples

The following example sets the speed to 1000BASE-T:

```
ciscoasa(config)# interface gigabitethernet0/1
ciscoasa(config-if)# speed 1000
ciscoasa(config-if)# duplex full
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
ciscoasa(config-if)# no shutdown
```

Related Commands

Command	Description
clear configure interface	Clears all configuration for an interface.
duplex	Sets the duplex mode.
interface	Configures an interface and enters interface configuration mode.
show interface	Displays the runtime status and statistics of interfaces.
show running-config interface	Shows the interface configuration.

spf-interval

To customize IS-IS throttling of shortest path first (SPF) calculations, use the **spf-interval** command in router isis configuration mode. To restore the default values, use the **no** form of this command.

spf-interval [**level-1 | level-2**] *spf-max-wait* [*spf-initial-wait spf-second-wait*]
no spf-interval [**level-1 | level-2**] *spf-max-wait* [*spf-initial-wait spf-second-wait*]

Syntax Description

level-1	(Optional) Apply intervals to Level-1 areas only.
level-2	(Optional) Apply intervals to Level-2 areas only.
spf-max-wait	Indicates the maximum interval (in seconds) between two consecutive SPF calculations. The range is from 1 to 120 seconds. The default is 10 seconds.
spf-initial-wait	(Optional) Indicates the initial SPF calculation delay (in milliseconds) after a topology change. The range is from 1 to 120000 milliseconds. The default is 5500 milliseconds (5.5 seconds).
spf-second-wait	(Optional) Indicates the hold time between the first and second SPF calculation (in milliseconds). The range is from 1 to 120000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

Command Default

spf-max-wait —10 seconds

spf-initial-wait —5500 milliseconds

spf-second-wait —5500 milliseconds

Command Modes

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

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

Command History

Release	Modification
9.6(1)	We introduced this command.

Usage Guidelines

SPF calculations are performed only when the topology changes. They are not performed when external routes change.

The **spf-interval** command controls how often the software performs the SPF calculation. The SPF calculation is processor-intensive. Therefore, it may be useful to limit how often the calculation is done, especially when

the area is large and the topology changes often. Increasing the SPF interval reduces the processor load of the router, but potentially slows down the rate of convergence.

The following description will help you determine whether to change the default values of this command:

- The *spf-initial-wait* argument indicates the initial wait time (in milliseconds) after a topology change before the first SPF calculation.
- The *spf-second-wait* argument indicates the interval (in milliseconds) between the first and second SPF calculation.
- Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the *spf-max-wait* interval specified; the SPF calculations are throttled or slowed down after the initial and second intervals. Once the *spf-max-wait* interval is reached, the wait interval continues at this interval until the network calms down.
- After the network calms down and there are no triggers for 2 times the *spf-max-wait* interval, fast behavior is restored (the initial wait time).

SPF throttling is not a dampening mechanism; that is, SPF throttling does not prevent SPF calculations or mark any route, interface, or router as down. SPF throttling simply increases the intervals between SPF calculations.

Examples

The following example configures intervals for SPF calculations, partial route calculation (PRC), and link-state packet (LSP) generation:

```
ciscoasa(config)# router isis
ciscoasa(config-router)# spf-interval 5 10 20
ciscoasa(config-router)# prc-interval 5 10 20
ciscoasa(config-router)# lsp-gen-interval 2 50 100
```

Related Commands

Command	Description
advertise passive-only	Configures the ASA to advertise passive interfaces.
area-password	Configures an IS-IS area authentication password.
authentication key	Enables authentication for IS-IS globally.
authentication mode	Specifies the type of authentication mode used in IS-IS packets for the IS-IS instance globally.
authentication send-only	Configure the IS-IS instance globally to have authentication performed only on IS-IS packets being sent (not received).
clear isis	Clears IS-IS data structures.
default-information originate	Generates a default route into an IS-IS routing domain.
distance	Defines the administrative distance assigned to routes discovered by the IS-IS protocol.
domain-password	Configures an IS-IS domain authentication password.

Command	Description
fast-flood	Configures IS-IS LSPs to be full.
hello padding	Configures IS-IS hellos to the full MTU size.
hostname dynamic	Enables IS-IS dynamic hostname capability.
ignore-lsp-errors	Configures the ASA to ignore IS-IS LSPs that are received with internal checksum errors rather than purging the LSPs.
isis adjacency-filter	Filters the establishment of IS-IS adjacencies.
isis advertise-prefix	Advertises IS-IS prefixes of connected networks in LSP advertisements on an IS-IS interface.
isis authentication key	Enables authentication for an interface.
isis authentication mode	Specifies the type of authentication mode used in IS-IS packets for the IS-IS instance per interface
isis authentication send-only	Configure the IS-IS instance per interface to have authentication performed only on IS-IS packets being sent (not received).
isis circuit-type	Configures the type of adjacency used for the IS-IS.
isis csnp-interval	Configures the interval at which periodic CSNP packets are sent on broadcast interfaces.
isis hello-interval	Specifies the length of time between consecutive hello packets sent by IS-IS.
isis hello-multiplier	Specifies the number of IS-IS hello packets a neighbor must miss before the ASA declares the adjacency as down.
isis hello padding	Configures IS-IS hellos to the full MTU size per interface.
isis lsp-interval	Configures the time delay between successive IS-IS LSP transmissions per interface.
isis metric	Configures the value of an IS-IS metric.
isis password	Configures the authentication password for an interface.
isis priority	Configures the priority of designated ASAs on the interface.
isis protocol shutdown	Disables the IS-IS protocol per interface.
isis retransmit-interval	Configures the amount of time between retransmission of each IS-IS LSP on the interface.
isis retransmit-throttle-interval	Configures the amount of time between retransmissions of each IS-IS LSP on the interface.
isis tag	Sets a tag on the IP address configured for an interface when the IP prefix is put into an LSP.

Command	Description
is-type	Assigns the routing level for the IS-IS routing process.
log-adjacency-changes	Enables the ASA to generate a log message when an NLSP IS-IS adjacency changes state (up or down).
lsp-full suppress	Configures which routes are suppressed when the PDU becomes full.
lsp-gen-interval	Customizes IS-IS throttling of LSP generation.
lsp-refresh-interval	Sets the LSP refresh interval.
max-area-addresses	Configures additional manual addresses for an IS-IS area.
max-lsp-lifetime	Sets the maximum time that LSPs persist in the ASA's database without being refreshed.
maximum-paths	Configures multi-path load sharing for IS-IS.
metric	Globally changes the metric value for all IS-IS interfaces.
metric-style	Configures an ASA running IS-IS so that it generates and only accepts new-style, length, value objects (TLVs).
net	Specifies the NET for the routing process.
passive-interface	Configures a passive interface.
prc-interval	Customizes IS-IS throttling of PRCs.
protocol shutdown	Disables the IS-IS protocol globally so that it cannot form any adjacency on any interface and will clear the LSP database.
redistribute isis	Redistributes IS-IS routes specifically from Level 1 into Level 2 or from Level 2 into Level 1.
route priority high	Assigns a high priority to an IS-IS IP prefix.
router isis	Enables IS-IS routing.
set-attached-bit	Specifies constraints for when a Level 1-Level 2 router should set its attached bit.
set-overload-bit	Configures the ASA to signal other routers not to use it as an intermediate hop in their SPF calculations.
show clns	Shows CLNS-specific information.
show isis	Shows IS-IS information.
show route isis	Shows IS-IS routes.
spf-interval	Customizes IS-IS throttling of SPF calculations.
summary-address	Creates aggregate addresses for IS-IS.

split-dns

To enter a list of domains to be resolved through the split tunnel, use the **split-dns** command in group-policy configuration mode. To delete a list, use the **no** form of this command.

To delete all split tunneling domain lists, use the **no split-dns** command without arguments. This deletes all configured split tunneling domain lists, including a null list created by issuing the **split-dns none** command.

When there are no split tunneling domain lists, users inherit any that exist in the default group policy. To prevent users from inheriting such split tunneling domain lists, use the **split-dns none** command.

split-dns { **value** *domain-name1 domain-name2 domain-nameN* | **none** }
no split-dns [*domain-name1 domain-name2 domain-nameN*]

Syntax Description

value *domain-name*	Provides a domain name that the ASA resolves through the split tunnel.
none	Indicates that there is no split DNS list. Sets a split DNS list with a null value, thereby disallowing a split DNS list. Prevents inheriting a split DNS list from a default or specified group policy.

Command Default

Split DNS is disabled.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

Use a single space to separate each entry in the list of domains. There is no limit on the number of entries, but the entire string can be no longer than 492 characters. You can use only alphanumeric characters, hyphens (-), and periods (.).

The **no split-dns** command, when used without arguments, deletes all current values, including a null value created by issuing the **split-dns none** command.

Starting with version 3.0.4235, Secure Client supports true split DNS functionality for Windows platforms.

Examples

The following example shows how to configure the domains Domain1, Domain2, Domain3 and Domain4 to be resolved through split tunneling for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-dns value Domain1 Domain2 Domain3 Domain4
```

Related Commands

Command	Description
default-domain	Specifies a default domain name that the IPsec client uses the for DNS queries which omit the domain field.
split-dns	Provides a list of domains to be resolved through the split tunnel.
split-tunnel-network-list	Identifies the access list the ASA uses to distinguish which networks require tunneling.
split-tunnel-policy	Lets an IPsec client conditionally direct packets over an IPsec tunnel in encrypted form, or to a network interface in cleartext form

split-horizon

To reenable EIGRP split horizon, use the **split-horizon** command in interface configuration mode. To disable EIGRP split horizon, use the **no** form of this command.

split-horizon eigrp *as-number*
no split-horizon eigrp *as-number*

Syntax Description

as-number	The autonomous system number of the EIGRP routing process.

Command Default

The **split-horizon** command is enabled.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was added.
9.0(1)	Multiple context mode is supported.

Usage Guidelines

For networks that include links over X.25 packet-switched networks, you can use the **neighbor** command to defeat the split horizon feature. As an alternative, you can explicitly specify the **no split-horizon eigrp** command in your configuration. However, if you do so, you must similarly disable split horizon for all routers and access servers in any relevant multicast groups on that network.

In general, it is best that you not change the default state of split horizon unless you are certain that your application requires the change in order to properly advertise routes. If split horizon is disabled on a serial interface and that interface is attached to a packet-switched network, you must disable split horizon for all routers and access servers in any relevant multicast groups on that network.

Examples

The following example disables EIGRP split horizon on interface Ethernet0/0:

```
ciscoasa(config)# interface Ethernet0/0
ciscoasa(config-if)# no split-horizon eigrp 100
```

Related Commands

Command	Description
router eigrp	Creates an EIGRP routing process and enters configuration mode for that process.

split-tunnel-all-dns

To enable the Secure Client to the resolve all DNS addresses through the VPN tunnel, use the split-tunnel-all-dns command from group policy configuration mode.

To remove the command from the running configuration, use the no form of this command. This enables inheritance of the value from another group policy.

split-tunnel-all-dns { **disable | enable** }
no split-tunnel-all-dns [{ **disable | enable** }]

Syntax Description

disable (default)	The Secure Client sends DNS queries over the tunnel according to the split tunnel policy—tunnel all networks, tunnel networks specified in a network list, or exclude networks specified in a network list.
enable	The Secure Client resolves all DNS addresses through the VPN tunnel.

Command Default

The default is disabled.

Command Modes

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

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

Command History

Release	Modification
8.2(5)	This command was added.

Usage Guidelines

The split-tunnel-all-dns enable command applies to VPN connections using the SSL or IPsec/IKEv2 protocol, and instructs the Secure Client to resolve all DNS addresses through the VPN tunnel. If DNS resolution fails, the address remains unresolved and the Secure Client does not try to resolve the address through public DNS servers.

By default, this feature is disabled. The client sends DNS queries over the tunnel according to the split tunnel policy—tunnel all networks, tunnel networks specified in a network list, or exclude networks specified in a network list.

Examples

The following example configures the ASA to enable the Secure Client to resolve all DNS queries through the VPN tunnel:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-tunnel-all-dns enable
```

Related Commands

Command	Description
default-domain	Specifies a default domain name that the legacy IPsec (IKEv1) VPN client or the AnyConnect VPN Client (SSL) uses for DNS queries that omit the domain field.
split-dns	Provides a list of domains to be resolved through the split tunnel.
split-tunnel-network-list	Identifies the access list the ASA uses to distinguish networks that require tunneling and those that do not.
split-tunnel-policy	Lets a legacy VPN client (IPsec/IKEv1) or the AnyConnect VPN client (SSL) conditionally direct packets over a tunnel in encrypted form, or to a network interface in clear text form

split-tunnel-network-list

To create a network list for split tunneling, use the **split-tunnel-network-list** command in group-policy configuration mode. To delete a network list, use the **no** form of this command.

split-tunnel-network-list { **value** *access-list name* | **none** }
no split-tunnel-network-list value [*access-list name*]

Syntax Description

none	Indicates that there is no network list for split tunneling; the ASA tunnels all traffic. Sets a split tunneling network list with a null value, thereby disallowing split tunneling. Prevents inheriting a default split tunneling network list from a default or specified group policy.
value *access-list name*	Identifies an access list that enumerates the networks to tunnel or not tunnel.

Command Default

By default, there are no split tunneling network lists.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

The ASA makes split tunneling decisions on the basis of a network list, which is a standard ACL that consists of a list of addresses on the private network. Split tunneling network lists distinguish networks that require traffic to travel across the tunnel from those that do not require tunneling.

When there are no split tunneling network lists, users inherit any network lists that exist in the default or specified group policy. To prevent users from inheriting such network lists, use the **split-tunnel-network-list none** command.

To delete all split tunneling network lists, use the **no split-tunnel-network-list** command without arguments. This deletes all configured network lists, including a null list created by issuing the **split-tunnel-network-list none** command.

Note Starting with version 9.7(1), you can specify up to 1200 split networks. In prior releases, the limit is 200 networks.

Examples

The following example shows how to set a network list called FirstList for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-tunnel-network-list value FirstList
```

Related Commands

Command	Description
access-list	Creates an access list, or uses a downloadable access list.
default-domain	Specifies a default domain name that he IPsec client uses the for DNS queries that omit the domain field.
split-dns	Provides a list of domains to be resolved through the split tunnel.
split-tunnel-policy	Lets an IPsec client conditionally direct packets over an IPsec tunnel in encrypted form, or to a network interface in cleartext form.

split-tunnel-policy

To set a split tunneling policy, use the **split-tunnel-policy** command in group-policy configuration mode. To remove the split-tunnel-policy attribute from the running configuration, use the **no** form of this command.

split-tunnel-policy { **tunnelall | tunnelspecified | excludespecified** }
no split-tunnel-policy

Syntax Description

excludespecified	Defines a list of networks to which traffic goes in the clear. This feature is useful for remote users who want to access devices on their local network, such as printers, while they are connected to the corporate network through a tunnel. This option works with Secure Client only.
split-tunnel-policy	Indicates that you are setting rules for tunneling traffic.
tunnelall	Specifies that no traffic goes in the clear or to any other destination than the ASA. Remote users reach Internet networks through the corporate network and do not have access to local networks.
tunnelspecified	Tunnels all traffic from or to the specified networks. This option enables split tunneling. It lets you create a network list of addresses to tunnel. Data to all other addresses travels in the clear, and is routed by the remote user's Internet service provider.

Command Default

Split tunneling is disabled by default, which is **tunnelall**.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

Split tunneling is primarily a traffic management feature, not a security feature. In fact, for optimum security, we recommend that you not enable split tunneling.

This enables inheritance of a value for split tunneling from another group policy.

Split tunneling lets a remote-access VPN client conditionally direct packets over an IPsec or SSL tunnel in encrypted form, or to a network interface in cleartext form. With split-tunneling enabled, packets not bound for destinations on the other side of the IPsec or SSL VPN tunnel endpoint do not have to be encrypted, sent across the tunnel, decrypted, and then routed to a final destination.

Examples

The following example shows how to set a split tunneling policy of tunneling only specified networks for the group policy named FirstGroup:

```
ciscoasa
(config)#
 group-policy FirstGroup attributes
ciscoasa
(config-group-policy)#
 split-tunnel-policy tunnelspecified
```

Related Commands

Command	Description
default-domain	Specifies a default domain name that he IPsec client uses the for DNS queries that omit the domain field.
split-dns	Provides a list of domains to be resolved through the split tunnel.
split-tunnel-network-list none	Indicates that no access list exists for split tunneling. All traffic travels across the tunnel.
split-tunnel-network-list value	Identifies the access list the ASA uses to distinguish networks that require tunneling and those that do not.

spoof-server

To substitute a string for the server header field for HTTP protocol inspection, use the **spoof-server** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

spoof-server*string*
no spoof-server *string*

Syntax Description

string	String to substitute for the server header field. 82 characters maximum.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

WebVPN streams are not subject to the spoof-server comand.

Examples

The following example shows how to substitute a string for the server header field in an HTTP inspection policy map:

```
ciscoasa(config-pmap-p)# spoof-server
string
```

Related Commands

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

sq-period

To specify the interval between each successful posture validation in a NAC Framework session and the next query for changes in the host posture, use the **sq-period** command in nac-policy-nac-framework configuration mode. To remove the command from the NAC policy, use the **no** form of this command.

sq-period *seconds*
no sq-period [*seconds*]

Syntax Description

seconds	Number of seconds between each successful posture validation. The range is 30 to 1800.

Command Default

The default value is 300.

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
Nac-policy-nac-framework configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.3(0)	"nac-" removed from command name. Command moved from group-policy configuration mode to nac-policy-nac-framework configuration mode.
7.2(1)	This command was added.

Usage Guidelines

The ASA starts the status query timer after each successful posture validation and status query response. The expiration of this timer triggers a query for changes in the host posture, referred to as a *status query* .

Examples

The following example changes the value of the status query timer to 1800 seconds:

```
ciscoasa(config-nac-policy-nac-framework)# sq-period 1800
ciscoasa(config-nac-policy-nac-framework)
```

The following example removes the status query timer from the NAC Framework policy:

```
ciscoasa(config-nac-policy-nac-framework)# no sq-period
ciscoasa(config-nac-policy-nac-framework)
```

Related Commands

Command	Description
nac-policy	Creates and accesses a Cisco NAC policy, and specifies its type.

Command	Description
nac-settings	Assigns a NAC policy to a group policy.
eou timeout	Changes the number of seconds to wait after sending an EAP over UDP message to the remote host in a NAC Framework configuration.
reval-period	Specifies the interval between each successful posture validation in a NAC Framework session.
debug eap	Enables logging of Extensible Authentication Protocol events to debug NAC Framework messaging.

srv-id

To configure a uri-id in a reference-identity object, use the **uri-id** command in ca-reference-identity mode. To delete a uri-id in, use the **no** form of this command. You can access the *ca-reference-identity* mode by first entering the **crypto ca reference-identity** command to configure a reference-identity object..

srv-id*value*
no srv-id *value*

Syntax Description

value	Value of each reference-id.
srv-id	A subjectAltName entry of type otherName whose name form is SRVName as defined in RFC 4985. A SRV-ID identifier may contain both a domain name and an application service type. For example, a SRV-ID of "_imaps.example.net" would be split into a DNS domain name portion of "example.net" and an application service type portion of "imaps."

Command Default

No default behavior or values.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
ca-reference-identity	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.6(2)	We introduced this command.

Usage Guidelines

Once a reference identity has been created, the four identifier types and their associated values can be added or deleted from the reference identity.

The reference identifiers MAY contain information identifying the application service and MUST contain information identifying the DNS domain name.

Examples

The following example creates a reference-identity for a syslog server:

```
ciscoasa(config)# crypto ca reference-identity syslogServer
ciscoasa(config-ca-ref-identity)# dns-id syslog1-bxb.cisco.com
ciscoasa(config-ca-ref-identity)# cn-id syslog1-bxb.cisco.com
```

Related Commands

Command	Description
crypto ca reference-identity	Configures a reference identity object.

Command	Description
cn-id	Configures a Common Name Identifier in the reference-identity object.
dns-id	Configures and DNS domain name Identifier in a reference identity object.
uri-id	Configures a URI identifier in a reference identity object.
logging host	Configures a logging server that can use a reference-identity object for a secure connection.
call-home profile destination address http	Configures a Smart Call Home server that can use a reference-identity object for a secure connection.

ss7 variant

To identify the SS7 variant used in your network for M3UA inspection, use the **ss7 variant** command in parameters configuration mode. You can access the parameters configuration mode by first entering the **policy-map type inspect m3ua** command. Use the **no** form of this command to return to the default SS7 variant.

ss7 variant { **ITU | ANSI | Japan | China** }
no ss7 variant { **ITU | ANSI | Japan | China** }

Syntax Description

ITU	The ITU variant. This is the default.
ANSI	The ANSI variant.
Japan	The Japan variant.
China	The China variant.

Command Default

The default is the ITU SS7 variant.

Command Modes

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

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

Command History

Release	Modification
9.6(2)	This command was added.

Usage Guidelines

Use this command to identify the SS7 variant used in your network. After you configure the option and deploy an M3UA policy, you cannot change it unless you first remove the policy.

The variant determines the format of the point codes used in M3UA messages.

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

Examples

The following example sets the SS7 variant to ITU.

```
ciscoasa(config)# policy-map type inspect m3ua m3ua-map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# ss7 variant ITU
```

Related Commands

Commands	Description
inspect m3ua	Enables M3UA inspection.
match dpc	Matches the M3UA destination point code.
match opc	Matches the M3UA originating point code.
policy-map type inspect	Creates an inspection policy map.

ssh

To add SSH access to the ASA, use the **ssh** command in global configuration mode. To disable SSH access to the ASA, use the **no** form of this command.

ssh { *ip_address mask* | *ipv6_address/prefix* } *interface*
no ssh { *ip_address mask* | *ipv6_address/prefix* } *interface*

Syntax Description

interface	The ASA interface on which SSH is enabled. Specify any named interface. For bridge groups, specify the bridge group member interface. For VPN management access only (see the **management-access** command), specify the named BVI interface.
ip_address	IPv4 address of the host or network authorized to initiate an SSH connection to the ASA.
ipv6_address/prefix	The IPv6 address and prefix of the host or network authorized to initiate an SSH connection to the ASA.
mask	Network mask for *ip_address*.

Command Default

No default behaviors or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
8.4(2)	You can no longer connect to the ASA using SSH with the **pix** or **asa** username and the login password. To use SSH, you must configure AAA authentication using the **aaa authentication ssh console LOCAL** command (CLI) or **Configuration > Device Management > Users/AAA > AAA Access > Authentication** (ASDM); then define a local user by entering the **username** command (CLI) or choosing **Configuration > Device Management > Users/AAA > User Accounts** (ASDM). If you want to use a AAA server for authentication instead of the local database, we recommend also configuring local authentication as a backup method.
8.4(4.1), 9.1(2)	You can enable public key authentication for SSH connections to the ASA on a per-user basis with the **ssh authentication** command.

Release	Modification
9.1(2)	The SSH server implementation in the ASA now supports AES-CTR mode encryption.
9.1(7)/9.4(3)/9.5(3)/9.6(1)	You can configure encryption and integrity ciphers for SSH access using the **ssh cipher encryption** and **ssh cipher integrity** commands.
9.6(2)	The **aaa authentication ssh console LOCAL** command is required for **ssh authentication**. In Version 9.6(2) and later, you can create a **username** without any password defined, so you can require public key authentication only.
9.7(1)	If you have a directly-connected SSH management station, you can use a /31 subnet on the ASA and the host to create a point-to-point connection.
9.6(3)/9.8(1)	Separate authentication for users with SSH public key authentication and users with passwords. You no longer have to explicitly enable AAA SSH authentication (**aaa authentication ssh console)**; when you configure the **ssh authentication** command for a user, local authentication is enabled by default for users with this type of authentication. Moreover, when you explicitly configure AAA SSH authentication, this configuration only applies for usernames with passwords, and you can use any AAA server type.
9.9(2)	Virtual interfaces can now be specified.

Usage Guidelines

The ssh *ip_address* command specifies hosts or networks that are authorized to initiate an SSH connection to the ASA. You can have multiple **ssh** commands in the configuration.

Before you can begin using SSH to the ASA, you must generate a default RSA key using the **crypto key generate rsa** command.

To access the ASA interface for SSH access, you do not also need an access rule allowing the host IP address. You only need to configure SSH access according to this section.

SSH access to an interface other than the one from which you entered the ASA is not supported. For example, if your SSH host is located on the outside interface, you can only initiate a management connection directly to the outside interface. The only exception to this rule is through a VPN connection (see the **management-access** command).

The ASA allows a maximum of 5 concurrent SSH connections per context/single mode, with a maximum of 100 connections divided among all contexts.

The ASA supports the SSH remote shell functionality provided in SSH Version 2 and supports DES and 3DES ciphers.

The following SSH Version 2 features are not supported on the ASA:

- X11 forwarding
- Port forwarding
- SFTP support
- Kerberos and AFS ticket passing
- Data compression

To use SSH with a username and password, you must configure AAA authentication using the **aaa authentication ssh console LOCAL** command; then define a local user by entering the **username** command. If you want to use a AAA server for authentication instead of the local database, we recommend also configuring local authentication as a backup method.

To use SSH with a local **username** and public key authentication, configure the **ssh authentication** command. Only the local database is supported.

In Version 9.6(2) and 9.7(1), the **aaa authentication ssh console LOCAL** command is required for **ssh authentication**. In Version 9.6(2) and later, you can create a **username** without any password defined, so you can require public key authentication only.

Note Do not use the **username** command **nopassword** option to avoid having to create a username with a password; the **nopassword** option allows *any* password to be entered, not no password. If you configure the **aaa** command, then the **nopassword** option creates a security problem.

For 9.6(1) and earlier and for 9.6(3)/9.8(1) and later, you do not have to configure the **aaa authentication ssh console LOCAL** command; this command only applies to users with passwords, and you can specify any server type, not just LOCAL. For example, some users can use public key authentication using the local database, and other users can use passwords with RADIUS. If you do configure the **aaa authentication ssh console LOCAL** command, you can choose to log in with either the **username** password, or with the private key.

Examples

The following example shows how to generate RSA keys and let a host on the inside interface with an address of 192.168.1.2 access the ASA:

```
ciscoasa(config)# crypto key generate rsa modulus 1024
ciscoasa(config)# write memory
ciscoasa(config)# aaa authentication ssh console LOCAL

WARNING: local database is empty! Use 'username' command to define local users.
ciscoasa(config)# username exampleuser1 password examplepassword1 privilege 15
ciscoasa(config)# ssh 192.168.1.2 255.255.255.255 inside
ciscoasa(config)# ssh timeout 30
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
crypto key generate rsa	Generates RSA key pairs for identity certificates.
debug ssh	Displays debugging information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh scopy enable	Enables a secure copy server on the ASA.
ssh version	Restricts the ASA to using either SSH Version 1 or SSH Version 2.

ssh authentication

To enable SSH public key authentication on a per-user basis, use the **ssh authentication** command in username attributes mode. To disable public key authentication on a per-user basis, use the **no** form of this command.

ssh authentication { **pkf | publickey** [**nointeractive**] *key* [**hashed**] }
no ssh authentication { **pkf | publickey** [**nointeractive**] *key* [**hashed**] }

Syntax Description

hashed	When you view the key on the ASA using the **show running-config username** command, the key is encrypted using a SHA-256 hash. Even if you entered the key as **pkf** , the ASA hashes the key, and shows it as a hashed **publickey** . If you need to copy the key from **show** output, specify the **publickey** type with the **hashed** keyword.
key	The value of the key argument can be one of the following: • When the key argument is supplied and the hashed tag is not specified, the value of the key must be a Base 64 encoded public key that is generated by SSH key generation software that can generate ssh-rsa, ecdsa-sha2-nistp, or ssh-ed25519 raw keys (that is, with no certificates). After you submit the Base 64 encoded public key, that key is then hashed via SHA-256 and the corresponding 32-byte hash is used for all further comparisons. • When the key argument is supplied and the hashed tag is specified, the value of the key must have been previously hashed with SHA-256 and be 32 bytes long, with each byte separated by a colon (for parsing purposes).
nointeractive	The **nointeractive** option suppresses all prompts when importing an SSH public key file formatted key. This noninteractive data entry mode is only intended for ASDM use.
pkf	For a **pkf** key, you are prompted to paste in a PKF formatted key, up to 4096 bits. Use this format for keys that are too large to paste inline in Base64 format. For example, you can generate a 4096-bit key using ssh keygen, then convert it to PKF, and use the **pkf** keyword to be prompted for the key. **Note** You can use the **pkf** option with failover, but the PKF key is not automatically replicated to the standby system. You must enter the **write standby** command to synchronize the PKF key.
publickey	For a **publickey** , the *key* is a Base64-encoded public key. You can generate the key using any SSH key generation software (such as ssh keygen) that can generate SSH-RSA raw keys (with no certificates).

Command Default

No default behaviors or values.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Username attributes	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
8.4(4.1), 9.1(2)	This command was added. *This feature is not available in 8.5(1), 8.6(1), 8.7(1), 9.0(1), 9.0(2), or 9.1(1).*
9.1(2)	We added the **pkf** keyword and support for keys up to 4096-bits.
9.6(2)	The **aaa authentication ssh console LOCAL** command is required for **ssh authentication** . In Version 9.6(2) and later, you can create a **username** without any password defined, so you can require public key authentication only.
9.6(3)/9.8(1)	Separate authentication for users with SSH public key authentication and users with passwords. You no longer have to explicitly enable AAA SSH authentication (**aaa authentication ssh console)** ; when you configure the **ssh authentication** command for a user, local authentication is enabled by default for users with this type of authentication. Moreover, when you explicitly configure AAA SSH authentication, this configuration only applies for usernames with passwords, and you can use any AAA server type.
9.16(1)	Support for EdDSA and ECDSA keys was added.

Usage Guidelines

For a local **username** , you can enable public key authentication instead of password authentication. You can generate a public key/private key pair using any SSH key generation software (such as ssh keygen) that can generate ssh-rsa, ecdsa-sha2-nistp, or ssh-ed25519 raw keys (with no certificates). Use the **ssh authentication** command to enter the public key on the ASA. The SSH client then uses the private key (and the passphrase you used to create the key pair) to connect to the ASA.

Only the local database is supported.

When you save the configuration, the hashed key value is saved to the configuration and used when the ASA is rebooted.

In Version 9.6(2) and 9.7(1), the **aaa authentication ssh console LOCAL** command is required for **ssh authentication** . In Version 9.6(2) and later, you can create a **username** without any password defined, so you can require public key authentication only.

Note Do not use the **username** command **nopassword** option to avoid having to create a username with a password; the **nopassword** option allows *any* password to be entered, not no password. If you configure the **aaa** command, then the **nopassword** option creates a security problem.

For 9.6(1) and earlier and for 9.6(3)/9.8(1) and later, you do not have to configure the **aaa authentication ssh console LOCAL** command; this command only applies to users with passwords, and you can specify any server type, not just LOCAL. For example, some users can use public key authentication using the local database, and other users can use passwords with RADIUS. If you do configure the **aaa authentication ssh console LOCAL** command, you can choose to log in with either the **username** password, or with the private key.

Examples

The following example shows how to authenticate using a PKF formatted key:

```
ciscoasa(config)# crypto key generate eddsa edwards-curve ed25519
ciscoasa(config)# write memory
ciscoasa(config)# username deanwinchester password examplepassword1 privilege 15
ciscoasa(config)# username deanwinchester attributes
ciscoasa(config-username)# ssh authentication pkf
Enter an SSH public key formatted file.
End with the word "quit" on a line by itself:
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "256-bit ED25519, converted by dean@dwinchester-mac from "
AAAAC3NzaC1lZDI1NTE5AAAAIDmIeTNfEOnuH0094p1MKX80fW2O216g4trnf7gwWe5Q
---- END SSH2 PUBLIC KEY ----
quit
INFO: Import of an SSH public key formatted file SUCCEEDED.
ciscoasa(config-username)# aaa authentication ssh console LOCAL
ciscoasa(config)# ssh 192.168.1.2 255.255.255.255 inside
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
debug ssh	Displays debugging information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh version	Restricts the ASA to using either SSH Version 1 or SSH Version 2.

ssh cipher encryption

Users can select encryption and integrity algorithms when configuring SSH access. For fine grain control over the SSH cipher encryption algorithms, use the **ssh cipher encryption** command in global configuration mode. Predefined levels are available, which correspond to particular sets of algorithms. Also, you can define a custom list by specifying multiple colon-delimited algorithms. To restore the default, use the **no** form of this command.

ssh cipher encryption { **all | fips | high | low | medium | custom** *encryption_1* [**:** *encryption_2* [**:** *...encryption_n*]] }
no ssh cipher encryption { **all | fips | high | low | medium | custom** *encryption_1* [**:** *encryption_2* [**:** *...encryption_n*]] }

Syntax Description

all	Specifies that all encryption algorithms are accepted.
custom *encryption_1* [**:** *encryption_2* [**:** *... encryption_n*]]	Specifies a custom set of encryption algorithms. Enter the **show ssh ciphers** command to view all available encryption algorithms. For example: **custom 3des-cbc:aes192-cbc:aes256-ctr**
fips	Specifies only FIPS-compliant encryption algorithms
high	Specifies only high strength encryption algorithms.
low	Specifies low, medium, and high strength encryption algorithms.
medium	Specifies the medium and high strength encryption algorithms.

Command Default

Medium is the default.

Command Modes

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

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

Command History

Release	Modification
9.1(7)/9.4(3)/9.5(3)/9.6(1)	This command was added.
9.16(1)	We added the **chacha20-poly1305@openssh.com** and **aes128-gcm@openssh.com** algorithms.

Usage Guidelines

This command is used with the **ssh cipher integrity** command. For encryption algorithms, the following values are possible:

- all—3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes128-gcm@openssh.com chacha20-poly1305@openssh.com aes192-ctr aes256-ctr
- fips—aes128-cbc aes256-cbc aes128-gcm@openssh.com
- high—aes256-cbc aes128-gcm@openssh.com chacha20-poly1305@openssh.com aes256-ctr
- low—3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr
- medium—3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr

Note If FIPS mode is enabled, then only the FIPS encryption and integrity algorithms are allowed.

Optionally, some of the algorithms can be deselected. When FIPS mode is enabled, the intersection of the currently configured algorithms and the FIPS-compliant algorithms is calculated. If not NULL, the resulting configuration is used. If NULL, then the default FIPS-compliant algorithms are used.

The performance of secure copy depends partly on the encryption cipher used. If you choose the medium cipher set, the ASA negotiates one of the following algorithms in order: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr. If the first algorithm proposed (3des-cbc) is chosen, then the performance is much slower than a more efficient algorithm such as aes128-cbc. To change the proposed ciphers, use the **ssh cipher encryption** command; for example, **ssh cipher encryption custom aes128-cbc** .

Examples

The following example shows the configuration of some custom SSH encryption algorithms:

```
ciscoasa(config)# ssh cipher encryption custom 3des-cbc:aes128-cbc:aes192-cbc
```

Related Commands

Command	Description
show ssh	Displays the configured ciphers.
show ssh ciphers	Displays the available cipher algorithms.
ssh cipher integrity	Specifies configured SSH cipher integrity algorithms.

ssh cipher integrity

Users can select encryption and integrity cipher modes when configuring SSH access. For fine grain control over the SSH cipher integrity algorithms, use the **ssh cipher integrity** command in global configuration mode. Pre-defined levels are available, which correspond to particular sets of algorithms. Also, a custom list can be defined by specifying multiple colon delimited algorithms. To restore the default, use the **no** form of this command.

ssh cipher integrity { **all | fips | high | low | medium | custom** *algorithm_1* [**:** *algorithm_2* [**:** *...algorithm_n*]] }
no ssh cipher integrity { **all | fips | high | low | medium | custom** *algorithm_1* [**:** *algorithm_2* [**:** *...algorithm_n*]] }

Syntax Description

all	Specifies that all integrity algorithms are accepted.
custom *algorithm_1*[**:***algorithm_2*[**:***...algorithm_n*]]	Specifies a custom set of integrity algorithms. Enter the **show ssh ciphers** command to view all available integrity algorithms. For example: **custom hmac-sha1:hmac-sha1-96:hmac-md5-96**
fips	Specifies only FIPS-compliant integrity algorithms
high	Specifies only high strength integrity algorithms.
low	Specifies low, medium, and high strength integrity algorithms.
medium	Specifies the medium and high strength integrity algorithms.

Command Default

(9.12 and later) High is the default.

(9.10 and earlier) Medium is the default.

Command Modes

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

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

Command History

Release	Modification
9.1(7)/9.4(3)/9.5(3)/9.6(1)	This command was added.

Release	Modification
9.12(1)	We added HMAC-SHA256 integrity cipher support. The default is now the high security set of ciphers (hmac-sha1 and hmac-sha2-256). The former default was the medium set.
9.13(1)	The following values of integrity algorithms are considered as insecure and deprecated: • all —hmac-sha1-96, hmac-md5, hmac-md5-96, hmac-sha2-256 • low — hmac-sha1-96 , hmac-md5, hmac-md5-96, hmac-sha2-256 • medium— hmac-sha1-96 The above values will be removed from later release.

Usage Guidelines

This command is used with the **ssh cipher encryption** command. For integrity algorithms, the following values are possible:

- all—hmac-sha1, hmac-sha1-96(Deprecated), hmac-md5(Deprecated), hmac-md5-96(Deprecated), hmac-sha2-256(Deprecated)
- fips—hmac-sha1, hmac-sha2-256
- high—hmac-sha1, hmac-sha2-256
- low—hmac-sha1, hmac-sha1-96(Deprecated), hmac-md5(Deprecated), hmac-md5-96(Deprecated), hmac-sha2-256(Deprecated)
- medium—hmac-sha1, hmac-sha1-96(Deprecated), hmac-md5, hmac-md5-96, hmac-sha2-256

Note If FIPS mode is enabled, then only the FIPS encryption and integrity algorithms are allowed.

Optionally, some of the algorithms can be deselected. When FIPS mode is enabled, the intersection of the currently configured algorithms and the FIPS-compliant algorithms is calculated. If not NULL, the resulting configuration is used. If NULL, then the default FIPS-compliant algorithms are used.

Examples

The following example shows the configuration of some custom SSH integrity algorithms:

```
ciscoasa(config)# ssh cipher integrity custom hmac-sha1-96:hmac-md5
```

Related Commands

Command	Description
show ssh	Displays the configured ciphers.
show ssh ciphers	Displays the available cipher algorithms.
ssh cipher encryption	Specifies configured SSH cipher encryption algorithms.

ssh disconnect

To disconnect an active SSH session, use the **ssh disconnect** command in privileged EXEC mode.

ssh disconnect *session_id*

Syntax Description

session_id	Disconnects the SSH session specified by the ID number.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

You must specify a session ID. Use the **show ssh sessions** command to obtain the ID of the SSH session you want to disconnect.

Examples

The following example shows an SSH session being disconnected:

```
ciscoasa# show ssh sessions
SID Client IP       Version Mode Encryption Hmac     State           Username
0   172.69.39.39    1.99    IN   aes128-cbc md5      SessionStarted  pat
                            OUT  aes128-cbc md5      SessionStarted  pat
1   172.23.56.236   1.5     -    3DES       -        SessionStarted  pat
2   172.69.39.29    1.99    IN   3des-cbc   sha1     SessionStarted  pat
                            OUT  3des-cbc   sha1     SessionStarted  pat
ciscoasa# ssh disconnect 2
ciscoasa# show ssh sessions
SID Client IP       Version Mode Encryption Hmac     State           Username
0   172.69.39.29    1.99    IN   aes128-cbc md5      SessionStarted  pat
                            OUT  aes128-cbc md5      SessionStarted  pat
1   172.23.56.236   1.5     -    3DES       -        SessionStarted  pat
```

Related Commands

Command	Description
show ssh sessions	Displays information about active SSH sessions to the ASA.

Command	Description
ssh timeout	Sets the timeout value for idle SSH sessions.

ssh key-exchange group

To set the SSH key exchange method, use the **ssh key-exchange group** command in global configuration mode. To restore the default, use the **no** form of this command.

ssh key-exchange group { **curve25519-sha256** | **dh-group14-sha1** | **dh-group14-sha256** | **ecdh-sha2-nistp256** }
no ssh key-exchange group

Syntax Description

curve25519-sha256	Uses Elliptic Curve 25519 SHA256 for the key exchange.
dh-group14-sha1	Uses Diffie-Hellman Group 14 SHA1 for the key exchange.
dh-group14-sha256	(Default) Uses Diffie-Hellman Group 14 SHA256 for the key exchange.
ecdh-sha2-nistp256	Uses Elliptic Curve Diffie-Hellman (ECDH) SHA2 NIST P-256 for the key exchange.

Command Default

(9.12 and later) By default, **dh-group14-sha256** is used.

(9.10 and earlier) By default, the **dh-group1-sha1** is used.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global configuration	• Yes	• Yes	• Yes	• Yes (Admin context only)	—

Command History

Release	Modification
9.16(1)	We added the **curve25519-sha256** and **ecdh-sha2-nistp256** options.
9.13(1)	The **dh-group1-sha1** option was deprecated and will be removed in a later release.
9.12(2)	Setting the SSH key exchange mode is restricted to the Admin context in multiple context mode.
9.12(1)	We added the **dh-group14-sha256** option, which is also now the default.
8.4(4.1), 9.1(2)	We introduced this command. *This feature is not available in 8.5(1), 8.6(1), 8.7(1), 9.0(1), 9.0(2), or 9.1(1).*

Usage Guidelines

A key exchanges like Diffie-Hellman (DH) provides a shared secret that cannot be determined by either party alone. The key exchange is combined with a signature and the host key to provide host authentication. This key-exchange method provides explicit server authentication. For more information about using DH key-exchange methods, see RFC 4253.

You must set the SSH key exchange in the Admin context; this setting is inherited by all other contexts.

Examples

The following example shows how to exchange keys using the DH Group 14 SHA1 key-exchange method:

```
ciscoasa(config)# ssh key-exchange group dh-group-14-sha1
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
crypto key generate rsa	Generates RSA key pairs for identity certificates.
debug ssh	Displays debugging information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh scopy enable	Enables a secure copy server on the ASA.
ssh version	Restricts the ASA to using either SSH Version 1 or SSH Version 2.

ssh key-exchange hostkey

If you do not want to use the default key order (EdDSA, ECDSA, and then RSA), identify the key pair you want to use wih the **ssh key-exchange hostkey** command in global configuration mode. To restore the default, use the **no** form of this command.

ssh key-exchange hostkey { **rsa** | **ecdsa** | **eddsa** }
no ssh key-exchange hostname

Syntax Description

ecdsa	Uses the ECDSA key only.
eddsa	Uses the EdDSA key only.
rsa	Uses the RSA key only. You must use a key size 2048 or higher. RSA key support will be removed in a later release, so we suggest using the other supported key types instead.

Command Default

By default, this command is disabled, and keys are tried in the following order: EdDSA, ECDSA, and then RSA.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global configuration	• Yes	• Yes	• Yes	• Yes (Admin context only)	—

Command History

Release	Modification
9.16(1)	We introduced this command.

Usage Guidelines

SSH tries keys in the following order: EdDSA, ECDSA, and then RSA. View the keys using the **show crypto key mypubkey** {**eddsa** | **ecdsa** | **rsa**} command. The keys used by SSH are called <Default-*type*-Key>. If you override the key order with the **ssh key-exchange hostkey rsa** command,you must use a key size 2048 or higher. For upgrade compatibility, smaller keys are only supported when you use the default key order. RSA key support will be removed in a later release, so we suggest using the other supported key types instead.

Examples

The following example forces use of the EdDSA key only:

```
ciscoasa(config)# ssh key-exchange hostkey eddsa
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
crypto key generate rsa	Generates RSA key pairs for identity certificates.
debug ssh	Displays debugging information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh key-exchange group	Sets the SSH key exchange method.
ssh scopy enable	Enables a secure copy server on the ASA.

ssh pubkey-chain

To manually add or delete SSH servers and their keys from the ASA database for the on-board Secure Copy (SCP) client, use the **ssh pubkey-chain** command in global configuration mode. To remove all host keys, use the **no** form of this command. To remove only a single server key, see the **server** command.

ssh pubkey-chain
no ssh pubkey-chain

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
9.1(5)	This command was added.

Usage Guidelines

You can copy files to and from the ASA using the on-board SCP client. The ASA stores the SSH host key for each SCP server to which it connects. You can manually add or delete servers and their keys from the ASA database if desired.

For each server (see the **server** command), you can specify the **key-string** (public key) or **key-hash** (hashed value) of the SSH host.

Examples

The following example adds an already hashed host key for the server at 10.86.94.170:

```
ciscoasa(config)# ssh pubkey-chain
ciscoasa(config-ssh-pubkey-chain)# server 10.86.94.170
ciscoasa(config-ssh-pubkey-server)# key-hash sha256
65:d9:9d:fe:1a:bc:61:aa:64:9d:fc:ee:99:87:38:df:a8:8e:d9:e9:ff:42:de:e8:8d:2d:bf:a9:2b:85:2e:19
```

The following example adds a host string key for the server at 10.7.8.9:

```
ciscoasa(config)# ssh pubkey-chain
ciscoasa(config-ssh-pubkey-chain)# server 10.7.8.9
ciscoasa(config-ssh-pubkey-server)# key-string
Enter the base 64 encoded RSA public key.
End with the word "exit" on a line by itself
```

```
ciscoasa(config-ssh-pubkey-server-string)# c1:b1:30:29:d7:b8:de:6c:97:77:10:d7:46:41:63:87
ciscoasa(config-ssh-pubkey-server-string)# exit
```

Related Commands

Command	Description
copy	Copies a file to or from the ASA.
key-hash	Enters a hashed SSH host key.
key-string	Enters a public SSH host key.
server	Adds an SSH server and host key to the ASA database.
ssh stricthostkeycheck	Enables SSH host key checking for the on-board Secure Copy (SCP) client.

ssh scopy enable

To enable Secure Copy (SCP) on the ASA, use the **ssh scopy enable** command in global configuration mode. To disable SCP, use the **no** form of this command.

ssh scopy enable
no ssh scopy enable

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
9.1(7)/9.4(3)/9.5(3)/9.6(1)	You can configure encryption and integrity ciphers for SSH access using the **ssh cipher encryption** and **ssh cipher integrity** commands.

Usage Guidelines

SCP is a server-only implementation; it will be able to accept and terminate connections for SCP but can not initiate them. The ASA has the following restrictions:

- There is no directory support in this implementation of SCP, limiting remote client access to the ASA internal files.
- There is no banner support when using SCP.
- SCP does not support wildcards.
- The ASA license must have the VPN-3DES-AES feature to support SSH version 2 connections.

Before initiating the file transfer, the ASA checks available Flash memory. If there is not enough available space, the ASA terminates the SCP connection. If you are overwriting a file in Flash memory, you still need to have enough free space for the file being copied to the ASA. The SCP process copies the file to a temporary file first, then copies the temporary file over the file being replaced. If you do not have enough space in Flash to hold the file being copied and the file being overwritten, the ASA terminates the SCP connection.

The performance of secure copy depends partly on the encryption cipher used. By default, the ASA negotiates one of the following algorithms in order: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr. If the first algorithm proposed (3des-cbc) is chosen, then the performance is much slower than a

more efficient algorithm such as aes128-cbc. To change the proposed ciphers, use the **ssh cipher encryption** command; for example, **ssh cipher encryption custom aes128-cbc**.

Examples

The following example shows how to configure the inside interface to accept SSH Version 2 connections from a management console with the IP address 10.1.1.1. The idle session timeout is set to 60 minutes and SCP is enabled.

```
ciscoasa(config)# ssh 10.1.1.1 255.255.255.0 inside
ciscoasa(config)# ssh version 2
ciscoasa(config)# ssh scopy enable
ciscoasa(config)# ssh timeout 60
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
debug ssh	Displays debug information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh	Allows SSH connectivity to the ASA from the specified client or network.
ssh version	Restricts the ASA to using either SSH Version 1 or SSH Version 2.

ssh stack ciscossh

To use the CiscoSSH stack, use the **ssh stack ciscossh** command in global configuration mode. To use the proprietary ASA SSH stack, use the **no** form of this command.

ssh stack ciscossh
no ssh stack ciscossh

Command Default

CiscoSSH stack is enabled by default.

Command Modes

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

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

Command History

Release	Modification
9.17(1)	This command was added.
9.19(1)	This command is now enabled by default.

Usage Guidelines

The ASA supports two SSH stacks for SSH connections: a proprietary SSH stack or the CiscoSSH stack. CiscoSSH is based on OpenSSH. Cisco SSH supports:

- FIPS compliance
- Regular updates, including updates from Cisco and the open source community

Note that the CiscoSSH stack does not support:

- SSH to a different interface over VPN (management-access)
- EdDSA key pair
- RSA key pair in FIPS mode

If you need these features, you should use the ASA SSH stack.

There is a small change to SCP functionality with the CiscoSSH stack: to use the ASA **copy** command to copy a file to or from an SCP server, you have to enable SSH access on the ASA for the SCP server subnet/host using the **ssh** command.

Examples

The following example shows how to disable the CiscoSSH stack.

```
ciscoasa(config)# no ssh stack ciscossh
ciscoasa(config)#
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
debug ssh	Displays debug information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh	Allows SSH connectivity to the ASA from the specified client or network.

ssh stricthostkeycheck

To enable SSH host key checking for the on-board Secure Copy (SCP) client, use the **ssh stricthostkeycheck** command in global configuration mode. To disable host key checking, use the **no** form of this command.

ssh stricthostkeycheck
no ssh stricthostkeycheck

Syntax Description

This command has no arguments or keywords.

Command Default

By default, this command is enabled.

Command Modes

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

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

Command History

Release	Modification
9.1(5)	This command was added.

Usage Guidelines

You can copy files to and from the ASA using the on-board SCP client. When this option is enabled, you are prompted to accept or reject the host key if it is not already stored on the ASA. When this option is disabled, the ASA accepts the host key automatically if it was not stored before.

Examples

The following example enables SSH host key checking:

```
ciscoasa# ssh stricthostkeycheck
ciscoasa# copy x scp://cisco@10.86.95.9/x
The authenticity of host '10.86.95.9 (10.86.95.9)' can't be established.
RSA key fingerprint is dc:2e:b3:e4:e1:b7:21:eb:24:e9:37:81:cf:bb:c3:2a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.86.95.9' (RSA) to the list of known hosts.
Source filename [x]?
Address or name of remote host [10.86.95.9]?
Destination username [cisco]?
Destination password []? cisco123
Destination filename [x]?
```

Related Commands

Command	Description
copy	Copies a file to or from the ASA.

Command	Description
key-hash	Enters a hashed SSH host key.
key-string	Enters a public SSH host key.
server	Adds an SSH server and host key to the ASA database.
ssh pubkey-chain	Manually adds or deletes servers and their keys from the ASA database.

ssh timeout

To change the default SSH session idle timeout value, use the **ssh timeout** command in global configuration mode. To restore the default timeout value, use the **no** form of this command.

ssh timeout *number*
no ssh timeout

Syntax Description

number	Specifies the duration in minutes that an SSH session can remain inactive before being disconnected. Valid values are from 1 to 60 minutes.

Command Default

The default session timeout value is 5 minutes.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

The ssh timeout command specifies the duration in minutes that a session can be idle before being disconnected. The default duration is 5 minutes.

Examples

The following example shows how to configure the inside interface to accept only SSH version 2 connections from a management console with the IP address 10.1.1.1. The idle session timeout is set to 60 minutes and SCP is enabled.

```
ciscoasa(config)# ssh 10.1.1.1 255.255.255.0 inside
ciscoasa(config)# ssh version 2
ciscoasa(config)# ssh copy enable
ciscoasa(config)# ssh timeout 60
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
show running-config ssh	Displays the current SSH commands in the running configuration.

Command	Description
show ssh sessions	Displays information about active SSH sessions to the ASA.
ssh disconnect	Disconnects an active SSH session.

ssh version (Deprecated)

To restrict the version of SSH accepted by the ASA, use the **ssh version** command in global configuration mode. To restore the default value, use the **no** form of this command. Only Version 2 is supported.

ssh version 2
no ssh version 2

Syntax Description

2	Specifies that only SSH Version 2 connections are supported.

Command Default

Version 2 is the default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
9.9(1)	Version **1** was deprecated, and the **1** keyword will be removed in a later release. The default setting was also changed from **ssh version 1 2** to **ssh version 2** only.
9.16(1)	This command was removed.

Usage Guidelines

You should only set the SSH version to version 2.

Examples

The following example shows how to configure the inside interface to accept SSH Version 2 connections from a management console with the IP address 10.1.1.1. The idle session timeout is set to 60 minutes and SCP is enabled.

```
ciscoasa(config)# ssh 10.1.1.1 255.255.255.0 inside
ciscoasa(config)# ssh version 2
ciscoasa(config)# ssh copy enable
ciscoasa(config)# ssh timeout 60
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.

Command	Description
debug ssh	Displays debug information and error messages for SSH commands.
show running-config ssh	Displays the current SSH commands in the running configuration.
ssh	Allows SSH connectivity to the ASA from the specified client or network.

ssl certificate-authentication

To enable client certificate authentication for backwards compatibility for versions previous to 8.2(1), use the **ssl certificate-authentication** command in global configuration mode. To disable ssl certificate authentication, use the **no** version of this command.

ssl certificate-authentication [**fca-timeout** *timeout-in minutes*] **interface** *interface-name* **port** *port-number*
no ssl certificate-authentication [**fca-timeout** *timeout-in minutes*] **interface** *interface-name* **port** *port-number*

Syntax Description

fca-timeout	Forced certificate authentication timeout value in minutes.
interface-name	The name of the selected interface, such as inside, management, and outside.
port-number	The TCP port number, an integer in the range 1-65535.

Command Default

This feature is disabled by default.

Command Modes

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

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

Command History

Release	Modification
8.0(3)	This command was added.
8.2(1)	This command is no longer needed, but the ASA retains it for downgrading to previous versions.

Usage Guidelines

This command replaces the deprecated **http authentication-certificate** command.

Examples

The following example shows how to configure the ASA to use the SSL certificate authentication feature:

```
ciscoasa
(config)#
 ssl certificate-authentication interface inside port 330
```

Related Commands

Command	Description
show running-config ssl	Displays the current set of configured SSL commands.

ssl cipher

To specify the encryption algorithms for the SSL, DTLS, and TLS protocols, use the **ssl cipher** command in global configuration mode. To restore the default, which is the complete set of encryption algorithms, use the **no** form of this command.

ssl cipher *version* [*level* | **custom** *"string"*]
no ssl cipher *version* [*level* | **custom** *"string"*]

Syntax Description

custom *string*	Allows full control of the cipher suite using OpenSSL cipher definition strings.
level	Specifies the strength of the cipher and indicates the minimum level of ciphers that are supported. Valid values in increasing order of strength are: • **all**—Includes all ciphers, including NULL-SHA. • **low**—Includes all ciphers except NULL-SHA. • **medium**—Includes all ciphers except NULL-SHA, DES-CBC-SHA, and RC4-MD5. • **fips**—Includes all FIPS-compliant ciphers (excludes NULL-SHA, DES-CBC-SHA, RC4-MD5, RC4-SHA, and DES-CBC3-SHA. • **high** (applies only to TLSv1.2)—Includes only AES-256 with SHA-2 ciphers.
version	Specifies the SSL, DTLS, or TLS protocol version. Supported versions include: • **default**—The set of ciphers for outbound connections. • **dtlsv1**—The ciphers for DTLSv1 inbound connections. • dtlsv1.2 -The ciphers for DTLSv1.2 inbound connections. • **tlsv1**—The ciphers for TLSv1 inbound connections. • **tlsv1.1**—The ciphers for TLSv1.1 inbound connections. • **tlsv1.2**—The ciphers for TLSv1.2 inbound connections.

Command Default

The default is **medium** for all protocol versions.

Command Modes

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

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

Command History

Release	Modification
9.16(1)	Removed support for DES configuration on enabling strong crypto licensing because the DES is considered to be a weak cipher. If DES is configured when a strong licensing is enabled, DES is converted to strong cipher, AES.
9.12(1)	Removed NULL-SHA from tlv1 supported ciphers on lina. Deprecated and removed ssl cipher tlsv1 all and ssl cipher tlsv1 custom NULL-SHA command.
9.10(1)	dtls1.2 option added.
9.4(1)	All SSLv3 configuration and support removed from the ASA.
9.3(2)	This command was added.

Usage Guidelines

This command replaced the **ssl encryption** command starting with ASA Version 9.3(2).

The recommended setting is **medium**. Using **high** may limit connectivity. Using **custom** may limit functionality if there are only a few ciphers configured. Restricting the default custom value limits outbound connectivity, including clustering.

For more information about ciphers using OpenSSL, see https://www.openssl.org/docs/apps/ciphers.html.

Use the **show ssl ciphers all** command to view the list of which ciphers support which versions. For example:

```
These are the ciphers for the given cipher level; not all ciphers are supported by all
versions of SSL/TLS.
These names can be used to create a custom cipher list:
  DHE-RSA-AES256-SHA256 (tlsv1.2)
  AES256-SHA256 (tlsv1.2)
  DHE-RSA-AES128-SHA256 (tlsv1.2)
  AES128-SHA256 (tlsv1.2)
  DHE-RSA-AES256-SHA (tlsv1, tlsv1.1, dtlsv1, tlsv1.2)
  AES256-SHA (sslv3, tlsv1, tlsv1.1, dtlsv1, tlsv1.2)
  DHE-RSA-AES128-SHA (tlsv1, tlsv1.1, dtlsv1, tlsv1.2)
  AES128-SHA (sslv3, tlsv1, tlsv1.1, dtlsv1, tlsv1.2)
  DES-CBC3-SHA (sslv3, tlsv1, tlsv1.1, dtlsv1, tlsv1.2)
  RC4-SHA (sslv3, tlsv1)
  RC4-MD5 (sslv3, tlsv1)
  DES-CBC-SHA (sslv3, tlsv1)
  NULL-SHA (sslv3, tlsv1)
```

The ASA specifies the order of priority for supported ciphers as:

Ciphers supported by TLSv1.2 (1-9)

1. DHE-RSA-AES256-SHA256

2. AES256-SHA256
3. DHE-RSA-AES128-SHA256
4. AES128-SHA256
5. DHE-RSA-AES256-SHA
6. AES256-SHA
7. DHE-RSA-AES128-SHA
8. AES128-SHA
9. DES-CBC3-SHA

Ciphers not supported by TLSv1.1 or TLSv1.2 (10-13)

1. RC4-SHA
2. RC4-MD5
3. DES-CBC-SHA
4. NULL-SHA

Examples

The following example shows how to configure the ASA to use TLSv1.1 FIPS-compliant ciphers:

```
ciscoasa
(config)#

ssl cipher tlsv1.1 fips
```

The following example shows how to configure the ASA to use TLSv1 custom ciphers:

```
ciscoasa
(config)#
 ssl cipher tlsv1 custom "RC4-SHA:ALL"
```

Related Commands

Command	Description
show running-config ssl	Displays the current set of configured SSL commands.
show ssl ciphers	Displays the list of supported ciphers.

ssl-client-certificate

To specify the certificate that the ASA should present to the LDAP server as the client certificate when using LDAPS, use the **ssl-client-certificate** command in aaa-server host configuration mode. To remove the certificate, use the **no** form of this command.

ssl-client-certificate *trustpoint_name*
no ssl-client-certificate *trustpoint_name*

Syntax Description

trustpoint_name	The name of the trustpoint that holds the certificate that the ASA should present to the LDAP server as the client certificate.

Command Default

No default.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Aaa-server host configuration (LDAP only)	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.18(1)	This command was added.

Usage Guidelines

This certificate is needed if you configure the LDAP server to verify the client certificate. You must also enable **ldap-over-ssl** for the server. If you do not configure a certificate, the ASA does not present one when the LDAP server asks for it. If an LDAP server is configured to require a peer certificate, the secure LDAP session will not complete and authentication/authorization requests will fail.

Example

The following example shows two LDAP servers using different trustpoints for client authentication.

```
asa(config)# show running-config aaa-server OPENLDAPS
aaa-server OPENLDAPS protocol ldap
aaa-server OPENLDAPS (manif) host 10.1.1.2
ldap-base-dn DC=example,DC=com
ldap-scope subtree
ldap-naming-attribute cn
ldap-login-password *****
ldap-login-dn cn=admin,dc=example,dc=com
ldap-over-ssl enable
ssl-client-certificate LDAPS_TP_1
```

```
server-type auto-detect
aaa-server OPENLDAPS (manif) host 10.2.2.5
ldap-base-dn DC=example,DC=com
ldap-scope subtree
ldap-naming-attribute cn
ldap-login-password *****
ldap-login-dn cn=admin,dc=example,dc=com
ldap-over-ssl enable
ssl-client-certificate LDAPS_TP_2
server-type auto-detect
```

Related Commands

Command	Description
ldap-over-ssl	Configures LDAPS as the communications protocol for the LDAP server.

ssl client-version

To specify the SSL/TLS protocol version that the ASA uses when acting as a client, use the **ssl client-version** command in global configuration mode. To revert to the default, use the **no** form of this command.

ssl client-version [**any | sslv3-only | tlsv1-only | sslv3 | tlsv1 | tlsv1.1 | tlsv1.2**]
no ssl client-version

Syntax Description

any	Transmits SSLv3 client hellos and negotiates SSLv3 (or greater).
sslv3	Transmits SSLv3 client hellos and negotiates SSLv3 (or greater).
sslv3-only	Transmits SSLv3 client hellos and negotiates SSLv3 (or greater). **Note** This option has been deprecated as of Version 9.3(2).
tlsv1	Transmits TLSv1 client hellos and negotiates TLSv1 (or greater).
tlsv1.1	Transmits TLSv1.1 client hellos and negotiates TLSv1.1 (or greater).
tlsv1.2	Transmits TLSv1.2 client hellos and negotiates TLSv1.2 (or greater).
tlsv1-only	Transmits TLSv1 client hellos and negotiates TLSv1 (or greater). **Note** This option has been deprecated as of Version 9.3(2).

Command Default

The default value is **tlsv1**.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
9.3(2)	SSLv3 has been deprecated. The default is now **tlsv1** instead of **any**. The **any** keyword has been deprecated.

Usage Guidelines

If you use the **any**, **sslv3**, or **sslv3-only** keywords, the command is accepted with the following warning.

```
WARNING: SSLv3 is deprecated. Use of TLSv1 or greater is recommended.
```

In the next major ASA release, these keywords will be removed from the ASA.

Examples

The following example shows how to configure the ASA to specify the SSLv3 protocol version when acting as an SSL client:

```
ciscoasa
(config)#
 ssl client-version any
```

Related Commands

Command	Description
clear config ssl	Removes all SSL commands from the configuration, reverting to the default values.
ssl encryption	Specifies the encryption algorithms that the SSL/TLS protocol uses.
show running-config ssl	Displays the current set of configured SSL commands.
ssl server-version	Specifies the minimum protocol version for which the ASA will negotiate an SSL/TLS connection.
ssl trust-point	Specifies the certificate trustpoint that represents the SSL certificate for an interface.

ssl dh-group

To specify the Diffie-Hellmann (DH) group to be used with DHE-RSA ciphers that are used by TLS, use the **ssl dh-group** command in global configuration mode. To return to the default, use the **no** form of this command.

ssl dh-group [**group1 | group2 | group5 | group14 | group24**]
no ssl dh-group [**group1 | group2 | group5 | group14 | group24**]

Syntax Description

group1	Configures DH group 1 (768-bit modulus).
group2	Configures DH group 2 (1024-bit modulus).
group5	Configures DH group 5 (1536-bit modulus).
group14	Configures DH group 14 (2048-bit modulus, 224-bit prime order subgroup).
group24	Configures DH group 24 (2048-bit modulus, 256-bit prime order subgroup).

Command Default

The default is DH group 14.

Command Modes

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

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

Command History

Release	Modification
9.16(1)	Support is remove for the command options group2, group5, and group24. Support was added for the command option group15.
9.13(1)	The group2 and group 5 command option was deprecated and will be removed in the later release.
9.3(2)	This command was added.

Usage Guidelines

Groups 1 and 2 are compatible with Java 7 and earlier versions. Groups 5, 14, and 24 are not compatible with Java 7. All groups are compatible with Java 8. Groups 14 and 24 are FIPS-compliant.

Examples

The following example shows how to configure the ASA to use a specific DH group:

```
ciscoasa
(config)#
 ssl dh-group group14
```

Related Commands

Command	Description
show running-config ssl	Displays the current set of configured SSL commands.

ssl ecdh-group

To specify the group to be used with ECDHE-ECDSA ciphers that are used by TLS, use the **ssl ecdh-group** command in global configuration mode. To return to the default, use the **no** form of this command.

ssl ecdh-group [**group19 | group20 | group21**]
no ssl ecdh-group [**group19 | group20 | group21**]

Syntax Description

group19	Configures group 19 (256-bit EC).
group20	Configures group 20 (384-bit EC).
group21	Configures group 21 (521-bit EC).

Command Default

The default is group 19.

Command Modes

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

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

Command History

Release	Modification
9.4(1)	This command was added.

Usage Guidelines

TLSv1.2 adds support for the following ciphers:

- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-GCM-SHA384
- AES256-GCM-SHA384
- ECDHE-ECDSA-AES256-SHA384
- ECDHE-RSA-AES256-SHA384
- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-GCM-SHA256
- RSA-AES128-GCM-SHA256

- ECDHE-ECDSA-AES128-SHA256
- ECDHE-RSA-AES128-SHA256

Note ECDSA and DHE ciphers are the highest priority.

Examples

The following example shows how to configure the ASA to use a specific DH group:

```
ciscoasa
(config)#
 ssl ecdh-group group21
```

Related Commands

Command	Description
show running-config ssl	Displays the current set of configured SSL commands.

ssl encryption (Deprecated)

Note The last supported release for this command was Version 9.3(1).

To specify the encryption algorithms for the SSL, DTLS, and TLS protocols, use the **ssl encryption** command in global configuration mode **.** To restore the default, which is the complete set of encryption algorithms, use the **no** form of this command.

ssl encryption [**3des-sha1**] [**aes128-sha1**] [**aes256-sha1**] [**des-sha1**] [**null-sha1**] [**rc4-md5**] [**rc4-sha1**] [**dhe-aes256-sha1**] [**dhe-aes128-sha1**]
no ssl encryption

Syntax Description

3des-sha1	Specifies triple DES 168-bit encryption with Secure Hash Algorithm 1 (FIPS-compliant).
aes128-sha1	Specifies triple AES 128-bit encryption with Secure Hash Algorithm 1 (FIPS-compliant).
aes256-sha1	Specifies triple AES 256-bit encryption with Secure Hash Algorithm 1 (FIPS-compliant).
dhe-aes128-sha1	Specifies AES 128-bit encryption cipher suites for Transport Layer Security (TLS) (FIPS-compliant).
dhe-aes256-sha1	Specifies AES 256-bit encryption cipher suites for Transport Layer Security (TLS) (FIPS-compliant).
des-sha1	Specifies DES 56-bit encryption with Secure Hash Algorithm 1.
null-sha1	Specifies null encryption with Secure Hash Algorithm 1. This setting enforces message integrity without confidentiality. **Caution** If you specify null-sha1, data is not encrypted.
rc4-md5	Specifies RC4 128-bit encryption with an MD5 hash function.
rc4-sha1	Specifies RC4 128-bit encryption with Secure Hash Algorithm 1.

Command Default

By default, the SSL encryption list on the ASA contains these algorithms in the following order:

1. RC4-SHA1
2. AES128-SHA1 (FIPS-compliant)
3. AES256-SHA1 (FIPS-compliant)
4. 3DES-SHA1 (FIPS-compliant)
5. DHE-AES256-SHA1 (FIPS-compliant)
6. DHE-AES128-SHA1 (FIPS-compliant)

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
9.1(2)	Support for SSL encryption using the DHE-AES128-SHA1 and DHE-AES256-SHA1 algorithms was added.
9.3(2)	This command was deprecated and has been replaced by the **ssl cipher** command.
9.12(1)	This command was removed.

Usage Guidelines

Issuing the command again overwrites the previous setting. The ASDM License tab reflects the maximum encryption that the license supports, not the value that you configure.

The ordering of the algorithms determines preference for their use. You can add or remove algorithms to meet the needs of your environment.

For FIPS-compliant Secure Client SSL connections, you must ensure a FIPS-compliant cipher is the first one specified in the list of SSL encryptions.

Several applications do not support DHE, so include at least one other SSL encryption method to ensure a cipher suite common to both.

Cryptographic operations use symmetric-key algorithms, as referenced in http://en.wikipedia.org/wiki/Symmetric-key_algorithm .

Examples

The following example shows how to configure the ASA to use the 3des-sha1 and des-sha1 encryption algorithms:

```
ciscoasa
(config)#
 ssl encryption 3des-sha1 des-sha1
```

Starting with ASA version 9.3(2)

The following examples show that this command has been deprecated and replaced by the **ssl cipher** command:

ciscoasa (config)# **ssl encryption ?**

```
configure mode commands/options:
This command is DEPRECATED, use 'ssl cipher' instead.
  3des-sha1        Indicate use of 3des-sha1 for ssl encryption
  aes128-sha1      Indicate use of aes128-sha1 for ssl encryption
  aes256-sha1      Indicate use of aes256-sha1 for ssl encryption
  des-sha1         Indicate use of des-sha1 for ssl encryption
  dhe-aes128-sha1  Indicate use of dhe-aes128-sha1 for ssl encryption
  dhe-aes256-sha1  Indicate use of dhe-aes256-sha1 for ssl encryption
  null-sha1        Indicate use of null-sha1 for ssl encryption (NOTE: Data is
                   NOT encrypted if this cipher is chosen)
  rc4-md5          Indicate use of rc4-md5 for ssl encryption
  rc4-sha1         Indicate use of rc4-sha1 for ssl encryption
```

ciscoasa (config)# **ssl encryption rc4-sha1 aes256-sha1 aes128-sha1**

```
WARNING: This command has been deprecated; use 'ssl cipher' instead.
INFO: Converting to: ssl cipher default custom "RC4-SHA:AES256-SHA:AES128-SHA"
INFO: Converting to: ssl cipher sslv3 custom "RC4-SHA:AES256-SHA:AES128-SHA"
INFO: Converting to: ssl cipher tlsv1 custom "RC4-SHA:AES256-SHA:AES128-SHA"
INFO: Converting to: ssl cipher dtlsv1 custom "RC4-SHA:AES256-SHA:AES128-SHA"
```

Related Commands

Command	Description
clear config ssl	Removes all SSL commands from the configuration, reverting to the default values.
show running-config ssl	Displays the current set of configured SSL commands.
ssl client-version	Specifies the SSL/TLS protocol version the ASA uses when acting as a client.
ssl server-version	Specifies the minimum protocol version for which the ASA will negotiate an SSL/TLS connection.
ssl trust-point	Specifies the certificate trust point that represents the SSL certificate for an interface.
ssl cipher	Specifies the encryption algorithms for the SSL, DTLS, and TLS protocols. **Note** Available as of the 9.3(2) release.

ssl server-version

To set the minimum protocol version for which the ASA will negotiate an SSL/TLS connection, use the **ssl server-version** command in global configuration mode. To revert to the default, any, use the **no** form of this command.

ssl server-version [[**tlsv1 | tlsv1.1 | tlsv1.2**] [**dtlsv1 | dtlsv1.2**]]
no ssl server-version

Syntax Description

tlsv1	Accepts SSLv2 client hellos and negotiates TLSv1 (or greater).
tlsv1.1	Accepts SSLv2 client hellos and negotiates TLSv1.1 (or greater).
tlsv1.2	Accepts SSLv2 client hellos and negotiates TLSv1.2 (or greater).
dtlsv1	Accepts DTLSv1 client hellos and negotiates DTLSv1 (or greater).
dtlsvv1.2	Accepts DTLSv1.2 client hellos and negotiates DTLSv1.2 (or greater). Specifying DTLSv1.2 tunnel use requires the specification of TLSv1.2 tunnel since it is the only valid option.

Command Default

The default values are **tlsv1** and **dtlsv1** .

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
9.3(2)	SSLv3 has been deprecated. The default is now **tlsv1** instead of **any** . The **any** keyword has been deprecated.
9.4(1)	All SSLv3 keywords have been removed from the ASA configuration, and SSLv3 support has been removed from the ASA. If you have SSLv3 enabled, a boot-time error will appear from the command with the SSLv3 option. The ASA will then revert to the default use of TLSv1.
9.10(1)	DTLS options provided now that DTLSv1.2 is supported.Previously assumed DTLS version 1 remains the default.

Examples

The following example shows how to configure the ASA to negotiate an SSL/TLS connection:

```
ciscoasa
(config)#
 ssl server-version tlsv1
```

The following example shows configuration and verification of set versions:

```
ciscoasa (config)# ssl server-version tlsv1.2 dtlsv1.2

ciscoasa (config)# sh run ssl
ssl server-version tlsv1.2 dtlsv1.2
ciscoasa (config)# no ssl server-version
ciscoasa (config)# sh run all ssl
ssl server-version tlsv1 dtlsv1
```

Related Commands

Command	Description
clear config ssl	Removes all SSL commands from the configuration, reverting to the default values.
show running-config ssl	Displays the current set of configured SSL commands.
ssl client-version	Specifies the SSL/TLS protocol version that the ASA uses when acting as a client.
ssl encryption	Specifies the encryption algorithms that the SSL/TLS protocol uses.
ssl trust-point	Specifies the certificate trustpoint that represents the SSL certificate for an interface.

ssl trust-point

To specify the certificate trustpoint that represents the SSL certificate for an interface, use the **ssl trust-point** command with the *interface* argument in global configuration mode. To remove an SSL trustpoint from the configuration that does not specify an interface, use the **no** form of this command. To remove an entry that does specify an interface, use the no **ssl trust-point** *name* [*interface*] form of the command.

ssl trust-point *name* [*interface* [**vpnlb-ip**] | **domain** *domain-name*]
no ssl trust-point *name* [*interface* [**vpnlb-ip**] | **domain** *domain-name*]

Syntax Description

domain *domain-name*	Associates this trustpoint with a particular domain name that is used to access this interface (for example, www.cisco.com).
interface	Specifies the name for the interface to which the trustpoint applies. The **nameif** command defines the name of the interface.
name	Specifies the name of the CA trustpoint as configured in the **crypto ca trustpoint** *name* command.
vpnlb-ip	Associates this trustpoint with the VPN load-balancing cluster IP address on this interface. Applies only to interfaces.

Command Default

The default is no trustpoint association. The ASA uses the default self-generated RSA key-pair certificate.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.
9.3(2)	The **domain** *domain-name* keyword-argument pair was added.

Usage Guidelines

If you do not specify an interface or domain, then this entry will represent the fallback trustpoint that is used on all interfaces that are not associated with a trustpoint of their own.

If you enter the **ssl trustpoint ?** command, the available configured trustpoints appear. If you enter the **ssl trust-point** *name* **?** command (for example, **ssl trust-point mysslcert ?**), the available configured interfaces for the trustpoint-SSL certificate association appear.

You may configure up to 16 trustpoints per interface.

Observe these guidelines when using this command:

- The value for *trustpoint* must be the name of the CA trustpoint as configured in the **crypto ca trustpoint** *name* command.
- The value for *interface* must be the *nameif* name of a previously configured interface.
- Removing a trustpoint also removes any **ssl trust-point** entries that reference that trustpoint.
- You can have one **ssl trust-point** entry for each interface and one that specifies no interfaces.
- A trustpoint configured with the **domain** keyword may apply to multiple interfaces (depending on how you connect).
- You may only have one **ssl trust-point** per *domain-name* value.
- You can reuse the same trustpoint for multiple entries.
- If the following error appears after you enter this command:

```
error:0B080074:x509 certificate routines:X509_check_private_key:key values
mismatch@x509_cmp.c:339
```

It means that a user has configured a new certificate to replace a previously configured certificate. No action is required.

- The certificates are chosen in the following order:
  - If a connection matches the value of the **domain** keyword, that certificate is chosen first. (**ssl trust-point** *name* **domain** *domain-name* command)
  - If a connection is made to the load-balancing address, the vpnlb-ip certificate is chosen. (**ssl trust-point** *name* **interface vpnlb-ip** command)
  - The certificate configured for the interface. (**ssl trust-point** *name* **interface** command)
  - The default certificate not associated with an interface. (**ssl trust-point** *name* command)
  - The ASA's self-signed, self-generated certificate.

Examples

The following example shows how to configure an SSL trustpoint called FirstTrust for the inside interface, and a trustpoint called DefaultTrust with no associated interface.

```
ciscoasa
(config)#
 ssl trust-point FirstTrust inside
ciscoasa
(config)#
 ssl trust-point DefaultTrust
```

The following example shows how to use the **no** form of the command to delete a trustpoint that has no associated interface:

```
ciscoasa
(config)#
 show running-configuration ssl
```

```
ssl trust-point FirstTrust inside
ssl trust-point DefaultTrust
ciscoasa(config)# no ssl trust-point
ciscoasa
(config)#
 show running-configuration ssl
ssl trust-point FirstTrust inside
```

The following example shows how to delete a trustpoint that does have an associated interface:

```
ciscoasa
(config)#
 show running-configuration ssl
ssl trust-point FirstTrust inside
ssl trust-point DefaultTrust
ciscoasa
(config)#
 no ssl trust-point FirstTrust inside
ciscoasa
(config)#
 show running-configuration ssl
ssl trust-point DefaultTrust
```

The following example shows how to assign a specific domain name to a configured trustpoint:

```
ciscoasa
(config)#
 ssl trust-point
            www-cert domain www.example.com
```

Related Commands

Command	Description
clear config ssl	Removes all SSL commands from the configuration, reverting to the default values.
show running-config ssl	Displays the current set of configured SSL commands.
ssl client-version	Specifies the SSL/TLS protocol version the ASA uses when acting as a client.
ssl encryption	Specifies the encryption algorithms that the SSL/TLS protocol uses.
ssl server-version	Specifies the minimum protocol version for which the ASA will negotiate an SSL/TLS connection.
show ssl	Displays SSL configuration statistics.

sso-server (Deprecated)

Note The last supported release for this command was Version 9.5(1).

To create a Single Sign-On (SSO) server for ASA user authentication, use the **sso-server** command in webvpn configuration mode. With this command, you must specify the SSO server type.

To remove an SSO server, use the **no** form of this command.

sso-server *name* **type** [*siteminder* | *saml-v1.1-post*]
no sso-server *name*

Note This command is required for SSO authentication.

Syntax Description

name	Specifies the name of the SSO server. Minimum of 4 characters and maximum of 31 characters.
saml-v1.1-post	Specifies that the ASA SSO server being configured is a SAML, Version 1.1, SSO server of the POST type.
siteminder	Specifies that the ASA SSO server being configured is a Computer Associates SiteMinder SSO server.
type	Specifies the type of SSO server. SiteMinder and SAML-V1.1-POST are the only types available.

Command Default

There is no default value or behavior.

Command Modes

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

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

Command History

Release	Modification
7.1(1)	This command was added.
9.5(2)	This command was deprecated due to support for SAML 2.0.

Usage Guidelines

Single sign-on support, available only for WebVPN, lets users access different secure services on different servers without entering a username and password more than once. The **sso-server** command lets you create an SSO server.

In the authentication, the ASA acts as a proxy for the WebVPN user to the SSO server. The ASA currently supports the SiteMinder SSO server (formerly Netegrity SiteMinder) and the SAML POST-type SSO server. Currently, the available arguments for the type option are restricted to *siteminder* or *saml-V1.1-post* .

Examples

The following example, entered in webvpn configuration mode, creates a SiteMinder-type SSO server named “example1”:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# sso-server example1 type siteminder
ciscoasa(config-webvpn-sso-siteminder)#
```

The following example, entered in webvpn configuration mode, creates a SAML, Version 1.1, POST-type SSO server named “example2”:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# sso-server example2 type saml-v1.1-post
ciscoasa(config-webvpn-sso-saml)#
```

Related Commands

Command	Description
assertion-consumer-url	Identifies the URL for the SAML-type SSO assertion consumer service.
issuer	Specifies the SAML-type SSO server’s security device name.
max-retry-attempts	Configures the number of times the ASA retries a failed SSO authentication attempt.
policy-server-secret	Creates a secret key used to encrypt authentication requests to a SiteMinder SSO server.
request-timeout	Specifies the number of seconds before a failed SSO authentication attempt times out.
show webvpn sso-server	Displays the operating statistics for an SSO server.
test sso-server	Tests an SSO server with a trial authentication request.
trustpoint	Specifies a trustpoint name that contains the certificate to use to sign the SAML-type browser assertion
web-agent-url	Specifies the SSO server URL to which the ASA makes SiteMinder SSO authentication requests.

sso-server value (group-policy webvpn) (Deprecated)

Note The last supported release for this command was Version 9.5(1).

To assign an SSO server to a group policy, use the **sso-server value** command in webvpn configuration mode available in group-policy configuration mode.

To remove the assignment and use the default policy, use the **no** form of this command.

To prevent inheriting the default policy, use the **sso-server none** command.

sso-server { **value** *name* | **none** }
[**no**] **sso-server value** *name*

Syntax Description

name	Specifies the name of the SSO server being assigned to the group policy.

Command Default

The default policy assigned to the group is DfltGrpPolicy.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Group-policy webvpn configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.1(1)	This command was added.
9.5(2)	This command was deprecated due to support for SAML 2.0.

Usage Guidelines

The **sso-server value** command, when entered in group-policy webvpn mode, lets you assign an SSO server to a group policy.

Single sign-on support, available only for WebVPN, lets users access different secure services on different servers without entering a username and password more than once. The ASA currently supports the SiteMinder-type of SSO server and the SAML POST-type SSO server.

This command applies to both types of SSO Servers.

Note Enter the same command, **sso-server value** , in username-webvpn configuration mode to assign SSO servers to user policies.

Examples

The following example commands create the group policy my-sso-grp-pol and assigns it to the SSO server named example:

```
ciscoasa(config)# group-policy my-sso-grp-pol internal
ciscoasa(config)# group-policy my-sso-grp-pol attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# sso-server value example
ciscoasa(config-group-webvpn)#
```

Related Commands

Command	Description
policy-server-secre t	Creates a secret key used to encrypt authentication requests to a SiteMinder SSO server.
show webvpn sso-server	Displays the operating statistics for all SSO servers configured on the security device.
sso-server	Creates a single sign-on server.
sso-server value (username webvpn)	Assigns an SSO server to a user policy.
web-agent-url	Specifies the SSO server URL to which the ASA makes SiteMinder-type SSO authentication requests.

sso-server value (username webvpn) (Deprecated)

Note The last supported release for this command was Version 9.5(1).

To assign an SSO server to a user policy, use the **sso-server value** command in webvpn configuration mode available in username configuration mode.

To remove an SSO server assignment for a user, use the **no** form of this command.

When a user policy inherits an unwanted SSO server assignment from a group policy, use the **sso-server none** command to remove the assignment.

sso-server { **value** *name* | **none** }
[**no**] **sso-server value** *name*

Syntax Description

name	Specifies the name of the SSO server being assigned to the user policy.

Command Default

The default is for the user policy to use the SSO server assignment in the group policy.

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
Username webvpn configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.1(1)	This command was added.
9.5(2)	This command was deprecated due to support for SAML 2.0.

Usage Guidelines

Single sign-on support, available only for WebVPN, lets users access different secure services on different servers without entering a username and password more than once. The ASA currently supports the SiteMinder-type of SSO server and the SAML POST-type SSO server.

This command applies to both types of SSO Servers.

The **sso-server value** command lets you assign an SSO server to a user policy.

Note Enter the same command, **sso-server value**, in group-webvpn configuration mode to assign SSO servers to group policies.

Examples

The following example commands assign the SSO server named my-sso-server to the user policy for a WebVPN user named Anyuser:

```
ciscoasa(config)# username Anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# sso-server value my-sso-server
ciscoasa(config-username-webvpn)#
```

Related Commands

Command	Description
policy-server-secret	Creates a secret key used to encrypt authentication requests to a SiteMinder SSO server.
show webvpn sso-server	Displays the operating statistics for all SSO servers configured on the security device.
sso-server	Creates a single sign-on server.
sso-server value (config-group-webvpn)	Assigns an SSO server to a group policy.
web-agent-url	Specifies the SSO server URL to which the ASA makes SiteMinder SSO authentication requests.

start-port

To configure the starting port for the port pool in the basic mapping rule in a Mapping Address and Port (MAP) domain, use the **start-port** command in MAP domain basic mapping rule configuration mode. Use the **no** form of this command to remove the ratio.

start-port*number*
no start-port *number*

Syntax Description

number	The first port in the port pool for the translated address. The port you specify must be a power of 2, from 1-32768 such as 1, 2, 4, 8, and so forth. If you want to exclude the well-known ports, start at 1024 or higher.

Command Default

No defaults.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
MAP domain basic mapping rule configuration mode.	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
9.13(1)	This command was introduced.

Usage Guidelines

The **start-port** and **share-ratio** commands in the basic mapping rule determine the starting port and number of ports in the pool used to translate addresses within a MAP domain.

Examples

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

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

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

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

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

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

```
ciscoasa(config-map-domain-bmr)# start-port 1024
ciscoasa(config-map-domain-bmr)# share-ratio 16
```

Related Commands

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

start-url

To enter the URL at which to retrieve an optional pre-login cookie, use the **start-url** command in aaa-server-host configuration mode. This is an SSO with HTTP Forms command.

start-url *string*

Note To configure SSO with the HTTP protocol correctly, you must have a thorough working knowledge of authentication and HTTP protocol exchanges.

Syntax Description

string The URL for an SSO server. The maximum URL length is 1024 characters.

Command Default

There is no default value or behavior.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Aaa-server-host configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.1(1)	This command was added.

Usage Guidelines

The WebVPN server of the ASA can use an HTTP POST request to submit a single sign-on authentication request to an authenticating web server. The authenticating web server may execute a pre-login sequence by sending a Set-Cookie header along with the login page content. You can discover this by connecting directly to the authenticating web server's login page with your browser. If the web server sets a cookie when the login page loads and if this cookie is relevant for the following login session, you must use the **start-url** command to enter the URL at which the cookie is retrieved. The actual login sequence starts after the pre-login cookie sequence with the form submission to the authenticating web server.

Note The **start-url** command is only required in the presence of the pre-login cookie exchange.

Examples

The following example, entered in aaa-server host configuration mode, specifies a URL for retrieving the pre-login cookie of https://example.com/east/Area.do?Page-Grp1:

```
ciscoasa(config)# aaa-server testgrp1 (inside) host example.com
ciscoasa(config-aaa-server-host)# start-url https://example.com/east/Area.do?Page=Grp1
ciscoasa(config-aaa-server-host)#
```

Related Commands

Command	Description
action-uri	Specifies a web server URI to receive a username and password for single sign-on authentication.
auth-cookie-name	Specifies a name for the authentication cookie.
hidden-parameter	Creates hidden parameters for exchange with the authenticating web server.
password-parameter	Specifies the name of the HTTP POST request parameter in which a user password must be submitted for SSO authentication.
user-parameter	Specifies the name of the HTTP POST request parameter in which a username must be submitted for SSO authentication.

state-checking

To enforce state checking for H.323, use the **state-checking** command in parameters configuration mode. To disable this feature, use the **no** form of this command.

state-checking [**h225 | ras**]
no state-checking [**h225 | ras**]

Syntax Description

h225	Enforces state checking for H.225.
ras	Enforces state checking for RAS.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to enforce state checking for RAS on an H.323 call:

```
ciscoasa(config)# policy-map type inspect h323 h323_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# state-checking ras
```

Related Commands

Command	Description
policy-map type inspect	Creates an inspection policy map.
show running-config policy-map	Display all current policy map configurations.

storage-url

To allow each context to use flash memory to store VPN packages, use the **storage-url** command in context configuration mode. To remove the storage space, use the **no** form of this command.

storage-url { **private | shared** } [**disk** *n* **:** /] *path* [*context_label*]
no storage-url { **private | shared** } [**disk** *n* **:** /] *path* [*context_label*]

Syntax Description

private	Assigns a private storage space to the context. You can specify one private storage space per context.
shared	Assigns a shared storage space to the context. You can specify one read-only shared storage space per context, but you can create multiple shared directories.
[**disk***n***:/**]*path*	Sets the path to the storage space. If you do not specify the disk number, the default is **disk0**. Under the specified path for the private storage space, the ASA creates a sub-directory named after the context. For example, for contextA if you specify **disk1:/private-storage** for the path, then the ASA creates a sub-directory for this context at disk1:/private-storage/contextA/. The ASA does not create context sub-directories for the shared storage space because it is a shared space for multiple contexts.
context_label	(Optional) You can name the path within the context with a *context_label* , so that the file system is not exposed to context administrators. For example, if you specify the *context_label* as **context**, then from within the context, this directory is called **context:**.

Command Default

If you do not specify the disk number, the default is **disk0**.

Command Modes

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

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

Command History

Release	Modification
9.6(2)	We introduced this command.

Usage Guidelines

Allow each context to use flash memory to store VPN packages, such as Secure Client, as well as providing storage for Secure Client and clientless SSL VPN portal customizations. Each context can use a private storage space as well as a shared read-only storage space. Note: Make sure the target directory is already present on the specified disk using the **mkdir** command.

You can specify one private storage space per context. You can read/write/delete from this directory within the context (as well as from the system execution space). To control how much disk space is allowed per context, see the **limit-resource storage** command.

To reduce duplication of common large files that can be ASA among all contexts, such as Secure Client packages, you can use the shared storage space. Only the system execution space can write and delete from the shared directory.

Examples

The following example creates a private directory and a shared directory, and assigns them to the admin context:

```
ciscoasa(config)# mkdir disk1:/private-storage
ciscoasa(config)# mkdir disk1:/shared-storage
ciscoasa(config)# context admin
ciscoasa(config-ctx)# storage-url private disk1:/private-storage context
ciscoasa(config-ctx)# storage-url shared disk1:/shared-storage shared
```

Related Commands

Command	Description
limit-resource storage	Controls how much disk space is allowed per context.

storage-key

To specify a storage key to protect the date stored between sessions, use the **storage-key** command in group-policy webvpn configuration mode. To remove this command from the configuration, use the **no** version of this command.

storage-key { **none | value** *string* }
nostorage-key

Syntax Description

string Specifies a string to use as the value of the storage key. This string can be up to 64 characters long.

Command Default

The default is **none**.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Group-policy webvpn configuration mode	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.

Usage Guidelines

While you can use any character except spaces in the storage key value, we recommend using only the standard alphanumeric character set: 0 through 9 and a through z.

Examples

The following example sets the storage key to the value abc123:

```
ciscoasa
(config)#

group-policy test attributes
ciscoasa
(config-group-policy)#
 webvpn
ciscoasa
(config-group-webvpn)#
storage-key value abc123
```

Related Commands

Command	Description
storage-objects	Configures storage objects for the data stored between sessions.

storage-objects

To specify which storage objects to use for the data stored between sessions, use the **storage-objects** command in group-policy webvpn configuration mode. To remove this command from the configuration, use the **no** version of this command.

storage-objects { **none | value** *string* }
no storage-objects

Syntax Description

string	Specifies the name of the storage objects. This string can be up to 64 characters long.

Command Default

The default is **none**.

Command Modes

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

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Group-policy webvpn configuration mode	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.

Usage Guidelines

While you can use any character except spaces and commas in the storage object name, we recommend using only the standard alphanumeric character set: 0 through 9 and a through z. Use a comma, with no space, to separate the names of storage objects in the string.

Examples

The following example sets the storage object names to cookies and xyz456:

```
ciscoasa
(config)#

group-policy test attributes
ciscoasa
(config-group-policy)#
 webvpn
ciscoasa
(config-group-webvpn)#
storage-object value cookies,xyz456
```

Related Commands

Command	Description
storage-key	Configures storage key to use for the data stored between sessions.
user-storage	Configures a location for storing user data between sessions

strict-asp-state

To enable strict M3UA application server process (ASP) state validation, use the **strict-asp-state** command in policy map parameters configuration mode. Use the **no** form of this command to remove the setting.

strict-asp-state
no strict-asp-state

Syntax Description

This command has no arguments or keywords.

Command Default

The default setting for this command is disabled.

Command Modes

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

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

Command History

Release	Modification
9.7(1)	This command was introduced.

Usage Guidelines

Use this command when configuring an M3UA inspection policy map.

If you enable strict application server process (ASP) state validation, the system maintains the ASP states of M3UA sessions and allows or drops ASP messages based on the validation result. If you do not enable strict ASP state validation, all ASP messages are forwarded uninspected.

Strict ASP state checking is required if you want stateful failover or if you want to operate within a cluster. However, strict ASP state checking works in Override mode only, it does not work if you are running in Loadsharing or Broadcast mode (per RFC 4666). The inspection assumes there is one and only one ASP per endpoint.

Examples

The following example enables strict checking for states and sessions:

```
ciscoasa(config)# policy-map type inspect m3ua m3ua-policy
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# strict-asp-state
```

Related Commands

Commands	Description
inspect m3ua	Enables M3UA inspection.

Commands	Description
policy-map type inspect m3ua	Creates an M3UA inspection policy map.

strict-diameter

To enable strict Diameter protocol conformance to RFC 6733, use the **strict-diameter** command in policy map parameters configuration mode. Use the **no** form of this command to remove the setting.

strict-diameter { **state | session** }
no strict-diameter { **state | session** }

Syntax Description

state	Enable state machine validation.
session	Enable session-related message validation.

Command Default

By default, inspection ensures that Diameter frames comply with the RFC, but state and session checking are not enabled.

Command Modes

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

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

Command History

Release	Modification
9.6(1)	This command was introduced.

Usage Guidelines

Use this command when configuring a Diameter inspection policy map.

These options enable strict compliance validation for states and sessions in addition to standard protocol conformance checks. You can enter the command twice to enable both state and session checking.

Examples

The following example enables strict checking for states and sessions:

```
ciscoasa(config)# policy-map type inspect diameter diameter-policy

ciscoasa(config-pmap)# parameters

ciscoasa(config-pmap-p)# strict-diameter state
ciscoasa(config-pmap-p)# strict-diameter session
```

Related Commands

Commands	Description
inspect diameter	Enables Diameter inspection.

Commands	Description
policy-map type inspect diameter	Creates a Diameter inspection policy map.

strict-header-validation

To enable strict validation of the header fields in the SIP messages according to RFC 3261, use the **strict-header-validation** command in parameters configuration mode. Parameters configuration mode is accessible from policy map configuration mode. To disable this feature, use the **no** form of this command.

strict-header-validation action { **drop | drop-connection | reset | log** } { **log** }
no strict-header-validation action { **drop | drop-connection | reset | log** } { **log** }

Syntax Description

drop	Drops the packet if validation occurs.
drop-connection	Drops the connection of a violation occurs.
reset	Resets the connection of a violation occurs.
log	Specifies standalone or additional log in case of violation. It can be associated to any of the actions.

Command Default

This command is disabled by default.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was added.

Examples

The following example shows how to enable strict validation of SIP header fields in a SIP inspection policy map:

```
ciscoasa(config)# policy-map type inspect sip sip_map
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# strict-header-validation action log
```

Related Commands

Command	Description
class	Identifies a class map name in the policy map.
class-map type inspect	Creates an inspection class map to match traffic specific to an application.

Command	Description
policy-map	Creates a Layer 3/4 policy map.
show running-config policy-map	Display all current policy map configurations.

strict-http

To allow forwarding of non-compliant HTTP traffic, use the **strict-http** command in HTTP map configuration mode, which is accessible using the **http-map** command. To reset this feature to its default behavior, use the **no** form of the command.

strict-http action { **allow | reset | drop** } [**log**]
no strict-http action action { **allow | reset | drop** } [**log**]

Syntax Description

action	The action taken when a message fails this command inspection.
allow	Allows the message.
drop	Closes the connection.
log	(Optional) Generate a syslog.
reset	Closes the connection with a TCP reset message to client and server.

Command Default

This command is enabled by default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

Although strict HTTP inspection cannot be disabled, the **strict-http action allow** command causes the ASA to allow forwarding of non-compliant HTTP traffic. This command overrides the default behavior, which is to deny forwarding of non-compliant HTTP traffic.

Examples

The following example allows forwarding of non-compliant HTTP traffic:

```
ciscoasa(config)# http-map inbound_http
ciscoasa(config-http-map)# strict-http allow
ciscoasa(config-http-map)#
```

Related Commands

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

strip-group

This command applies only to usernames received in the form user@realm. A realm is an administrative domain appended to a username with the "@" delimiter (juser@abc).

To enable or disable strip-group processing, use the **strip-group** command in tunnel-group general-attributes mode. The ASA selects the tunnel group for IPsec connections by obtaining the group name from the username presented by the VPN client. When strip-group processing is enabled, the ASA sends only the user part of the username for authorization/authentication. Otherwise (if disabled), the ASA sends the entire username including the realm.

To disable strip-group processing, use the **no** form of this command.

strip-group
no strip-group

Syntax Description

This command has no arguments or keywords.

Command Default

The default setting for this command is disabled.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

You can apply this attribute only to the IPsec remote access tunnel-type.

Note Because of a limitation of MSCHAPv2, you cannot perform tunnel group switching when MSCHAPv2 is used for PPP authentication. The hash computation during MSCHAPv2 is bound to the username string (such as user + delimit + group).

Examples

The following example configures a remote access tunnel group named "remotegrp" for type IPsec remote access, then enters general configuration mode, sets the tunnel group named "remotegrp" as the default group policy, and then enables strip group for that tunnel group:

```
ciscoasa(config)# tunnel-group remotegrp type IPsec_ra
ciscoasa(config)# tunnel-group remotegrp general
ciscoasa(config-tunnel-general)# default-group-policy remotegrp
ciscoasa(config-tunnel-general)# strip-group
```

Related Commands

Command	Description
clear-configure tunnel-group	Clears all configured tunnel groups.
group-delimiter	Enables group-name parsing and specifies the delimiter to be used when parsing group names from the user names that are received when tunnels are being negotiated.
show running-config tunnel group	Shows the tunnel group configuration for all tunnel groups or for a particular tunnel group.
tunnel-group general-attributes	Specifies the general attributes for the named tunnel-group.

strip-realm

To enable or disable strip-realm processing, use the **strip-realm** command in tunnel-group general-attributes configuration mode. Strip-realm processing removes the realm from the username when sending the username to the authentication or authorization server. A realm is an administrative domain appended to a username with the @ delimiter (username@realm). If the command is enabled, the ASA sends only the user part of the username authorization/authentication. Otherwise, the ASA sends the entire username.

To disable strip-realm processing, use the **no** form of this command.

strip-realm
no strip-realm

Syntax Description

This command has no arguments or keywords.

Command Default

The default setting for this command is disabled.

Command Modes

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

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

Command History

Release	Modification
7.0.1	This command was added.

Usage Guidelines

You can apply this attribute only to the IPsec remote access tunnel-type.

Examples

The following example configures a remote access tunnel group named “remotegrp” for type IPsec remote access, then enters general configuration mode, sets the tunnel group named “remotegrp” as the default group policy, and then enables strip realm for that tunnel group:

```
ciscoasa(config)# tunnel-group remotegrp type IPsec_ra
ciscoasa(config)# tunnel-group remotegrp general
ciscoasa(config-tunnel-general)# default-group-policy remotegrp
ciscoasa(config-tunnel-general)# strip-real
```