# sa - shov

# same-security-traffic

To permit communication between interfaces with equal security levels, or to allow traffic to enter and exit the same interface, use the **same-security-traffic** command in global configuration mode. To disable the same-security traffic, use the **no** form of this command.

**same-security-traffic permit** { **inter-interface | intra-interface** }
**no same-security-traffic permit** { **inter-interface | intra-interface** }

## Syntax Description

| | |
|---|---|
| **inter-interface** | Permits communication between different interfaces that have the same security level. |
| **intra-interface** | Permits communication in and out of the same interface. |

## 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 |
| Global configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 7.2(1) | The **intra-interface** keyword now allows all traffic to enter and exit the same interface, and not just IPsec traffic. |

## Usage Guidelines

Allowing communication between same security interfaces (enabled by the **same-security-traffic inter-interface** command) provides the following benefits:

- You can configure more than 101 communicating interfaces. If you use different levels for each interface, you can configure only one interface per level (0 to 100).
- You can allow traffic to flow freely between all same security interfaces without access lists.

The **same-security-traffic intra-interface** command lets traffic enter and exit the same interface, which is normally not allowed. This feature might be useful for VPN traffic that enters an interface, but is then routed out the same interface. The VPN traffic might be unencrypted in this case, or it might be re-encrypted for another VPN connection. For example, if you have a hub and spoke VPN network, where the Secure Firewall ASA is the hub, and remote VPN networks are spokes, for one spoke to communicate with another spoke, traffic must go into the ASA and then out again to the other spoke.

**Note** All traffic allowed by the **same-security-traffic intra-interface** command is still subject to firewall rules. Be careful not to create an asymmetric routing situation that can cause return traffic not to traverse the ASA.

## Examples

The following example shows how to enable the same-security interface communication:

```
ciscoasa(config)# same-security-traffic permit inter-interface
```

The following example shows how to enable traffic to enter and exit the same interface:

```
ciscoasa(config)# same-security-traffic permit intra-interface
```

## Related Commands

| Command | Description |
|---|---|
| show running-config same-security-traffic | Displays the **same-security-traffic** configuration. |

# sasl-mechanism

To specify a SASL (Simple Authentication and Security Layer) mechanism for authenticating an LDAP client to an LDAP server, use the **sasl-mechanism** command in aaa-server host configuration mode. The SASL authentication mechanism options are **digest-md5** and **kerberos** .

To disable an authentication mechanism, use the **no** form of this command.

**sasl-mechanism** { **digest-md5 | kerberos server-group-name** }
**no sasl-mechanism** { **digest-md5 | kerberos server-group-name** }

**Note** Because the ASA serves as a client proxy to the LDAP server for VPN users, the LDAP client referred to here is the ASA.

## Syntax Description

| | |
|---|---|
| *digest-md5* | The ASA responds with an MD5 value computed from the username and password. |
| kerberos | The ASA responds by sending the username and realm using the GSSAPI (Generic Security Services Application Programming Interface) Kerberos mechanism. |
| **server-group-name** | Specifies the Kerberos aaa-server group, up to 64 characters. |

## Command Default

No default behavior or values. The ASA passes the authentication parameters to the LDAP server in plain text.

**Note** We recommend that you secure LDAP communications with SSL using the **ldap-over-ssl** command if you have not configured SASL.

## 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 | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 7.1(1) | This command was added. |

## Usage Guidelines

Use this command to specify ASA authentication to an LDAP server using SASL mechanisms.

Both the ASA and the LDAP server can support multiple SASL authentication mechanisms. When negotiating SASL authentication, the ASA retrieves the list of SASL mechanisms configured on the server and sets the authentication mechanism to the strongest mechanism configured on both the ASA and the server. The Kerberos mechanism is stronger than the Digest-MD5 mechanism. To illustrate, if both the LDAP server and the ASA support both mechanisms, the ASA selects Kerberos, the stronger of the mechanisms.

When disabling the SASL mechanisms, you must enter a separate **no** command for each mechanism you want to disable because they are configured independently. Mechanisms that you do not specifically disable remain in effect. For example, you must enter both of the following commands to disable both SASL mechanisms:

**no sasl-mechanism digest-md5**
**no sasl-mechanism kerberos server-group-name**

## Examples

The following examples, entered in aaa-server host configuration mode, enable the SASL mechanisms for authentication to an LDAP server named ldapsvr1 with an IP address of 10.10.0.1. This example enables the SASL digest-md5 authentication mechanism:

```
ciscoasa(config)# aaa-server ldapsvr1 protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server ldapsvr1 host 10.10.0.1
ciscoasa(config-aaa-server-host)# sasl-mechanism digest-md5
```

The following example enables the SASL Kerberos authentication mechanism and specifies kerb-servr1 as the Kerberos AAA server:

```
ciscoasa(config)# aaa-server ldapsvr1 protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server ldapsvr1 host 10.10.0.1
ciscoasa(config-aaa-server-host)# sasl-mechanism kerberos kerbsvr1
```

## Related Commands

| Command | Description |
|---|---|
| **ldap-over-ssl** | Specifies that SSL secures the LDAP client-server connection. |
| server-type | Specifies the LDAP server vendor as either Microsoft or Sun. |
| **ldap attribute-map (global configuration mode)** | Creates and names an LDAP attribute map for mapping user-defined attribute names to Cisco LDAP attribute names. |

# saml idp

To add a new SAML IdP, use the saml idp command in webvpn configuration mode. To remove a SAML IdP, use the no form of this command.

**saml idp** *idp-entityID*
**no saml idp** *idp-entityID*

## Syntax Description

| | |
|---|---|
| base-URL | The Clientless VPN's base URL. It is used in SAML metadata that is provided to third-party IdPs, so that IdPs can redirect end users back to the ASA. |
| clock-skew <*value*> | Clock skew which will tolerate the NotBefore and NotOnOrAfter SAML assertions. By default the clock-skew should be disabled. The default value for the clock-skew is 1 second. The allowed range is 1 - 180 seconds. |
| **idp-entityID** | The entity ID of the SAML Idp you are configuring the ASA to use. |
| internal | Set this flag if the IdP is in the internal network. |
| signature | Enable or disable signature in a SAML request. |
| signature <value> | (Optional) Enable signature and use a specific method in a SAML request. |
| timeout assertion | Overrides NoOnOrAfter if the sum of NotBefore and timeout is earlier than NoOnOrAfter. |
| timeout-in-seconds | The SAML timeout value in seconds. By default, there is no SAML timeout. NotBefore and NotOnOrAfter in the assertion is used to determine the validity. |
| trustpoint [idp \| sp] <*trustpoint-name*> | The trustpoint idp contains the IdP certificate for ASA to verify SAML assertions.<br>The trustpoint-name is one of the existing trustpoint names.<br>The trustpoint sp contains the ASA (SP's) certificate for IdP to verify the ASA's signature or encrypt SAML assertion. |
| url [sign-in \| sign-out] <*value*> | The URL is the IdP's sign-in and sign-out URL.<br>The value of the URL for signing into the IdP. The url value must contain 4 to 2000 characters. |

## Command Default

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 |
| webvpn | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.5(2) | This command was added. |
| 9.7(1) | The internal attribute was added. |
| 9.8(1) | Added SHA2 support and the ability to specify a signature method in a SAML request. |

## Usage Guidelines

This command configures one or more third party SAML identity provider's settings. The IdP settings are not used until they are applied in a tunnel group.

The SAML IdP's sign-in url, sign-out url, signing certificate can be found on the vendor's website. You must create a trustpoint to hold the IdP's signing certificate. The trustpoint name will be used by trustpoint idp.

Creating an Idp in webvpn mode puts you into saml-idp sub-mode, where you configure the following settings for this Idp:

- url sign-in—URL to sign in to the Idp.
- url sign-out—URL for redirecting to when signing out of the IdP.
- signature—Enable or disable signature in SAML request. By default, the signature is disabled.
- signature <value>—Enable signature and specify the method as rsa-sha1, rsa-sha256, rsa-sha384, or rsa-sha512. By default the signature is disabled.
- time-out—SAML timeout value in seconds.
- base-url—URL is provided to third-party IdPs to redirect end-users back to the ASA. When base-url is not configured, the URL comes from the ASA's hostname and domain-name. For example, https://ssl-vpn.cisco.com as the base URL in show saml metadata when hostname is "ssl-vpn" and domain name is "cisco.com." If neither base-url or hostname/domain-name are configured, show saml metadata returns an error.
- trustpoint—Assigns an existing trustpoint based on the ASA (SP)'s or IDP certificate that the IdP can use to verify ASA's signature or encrypt SAML assertion.

## Examples

The following example shows how to define an Idp, and configure the Idp settings:

```
ciscoasa(config)# same-security-traffic permit inter-interface
ciscoasa(config-webvpn)# saml idp salesforce_idp

ciscoasa(config-webvpn-saml-idp)# url sign-in
 https://asa-dev-ed.my.salesforce.com/idp/endpoint/HttpRedirect

ciscoasa(config-webvpn-saml-idp)# url sign-out
 https://asa-dev-ed.my.salesforce.com/idp/endpoint/HttpRedirect
ciscoasa(config-webvpn-saml-idp)# trustpoint idp salesforce_trustpoint
ciscoasa(config-webvpn-saml-idp)# trustpoint sp asa_trustpoint
ciscoasa(config-webvpn)# saml idp feide_idp

ciscoasa(config-webvpn-saml-idp)# url sign-in
http://cisco.feide.no/simplesaml/saml2/idp/SSOService.php

ciscoasa(config-webvpn-saml-idp)# trustpoint idp feide_trustpoint
ciscoasa(config-webvpn-saml-idp)# trustpoint sp asa_trustpoint
ciscoasa(config-webvpn-saml-idp)# signature
```

```
ciscoasa(config-webvpn-saml-idp)# timeout assertion 120
ciscoasa(config-webvpn-saml-idp)# base-url https://ssl-vpn.cisco.com
```

## Related Commands

| Command | Description |
|---|---|
| authentication | Sets the authentication type for a tunnel group, such as saml. |
| identity-provider | Names this configuration of a third-party SAML identity provider in the ASA. |

# saml idp-trustpoint

To override the trustpoint IdP setting in the SAML IdP configuration, use the **saml idp-trustpoint** command in the webvpn tunnel group configuration mode. To remove the IdP trustpoint settings, use the no form of the command

**saml idp-trustpoint** *trustpoint_name* [ *trustpoint_name2* ]
**no saml idp-trustpoint** *name*

## Syntax Description

| | |
|---|---|
| *trustpoint_name* | Name of the IdP trustpoint. |
| *trustpoint_name2* | Name of the second IdP trustpoint. |

## Command Default

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 |
| config-tunnel-webvpn | • Yes | • Yes | • Yes | • No | — |

## Command History

| Release | Modification |
|---|---|
| 9.17(1) | This command was added. |
| 9.20(3) | *trustpoint_name2* variable was added. |

## Usage Guidelines

Existing ASA SAML configurations support only one IDP trustpoint for each configured SAML IDP. The **saml idp-trustpoint** command overrides the IdP settings to support the Microsoft Azure multiple application deployment scenario.

If the IdP trustpoint setting is present in the tunnel-group, the command overrides the trustpoint IdP setting in the IdP configuration, which is referenced by the **saml identity-provider** command in the tunnel group.

You can now configure two tunnel-group-specific IdP certificates. This feature lets you trust an old certificate as well as a new certificate, making migration to the new certificate easier. If your IdP, for example Azure IdP, has multiple applications but share the same SAML entity ID, and each application has its own certificate. You can use this command to override the main webvpn saml truspoint configuration.

## Examples

The following example shows how to override the IdP settings in trustpoint IdP configuration:

```
ciscoasa(config)# same-security-traffic permit inter-interface
ciscoasa(config-webvpn)# tunnel-group Sales webvpn-attributes
ciscoasa(config-tunnel-webvpn)# saml idp-trustpoint _SmartCallHome_ServerCA
```

```
_SmartCallHome_ServerCA2
```

## Related Commands

| Command | Description |
|---|---|
| identity-provider | Names configuration of a third-party SAML identity provider in the ASA. |

# saml identity-provider

Use this CLI in config-tunnel-webvpn mode to assign a SAML IdP to a tunnel group (connection profile)

**saml identity-provider** *name*
**no saml identity-provider** *name*

## Syntax Description

| | |
|---|---|
| **name** | The name of the SAML Idp you are configuring the ASA to use. |

## Command Default

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 |
| webvpn | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.5(2) | This command was added. |

## Usage Guidelines

This names this configuration of a third-party SAML identity provider in the ASA.

**Note** While adding the SAML identity provider name, if you get the error "ERROR: SAML configuration could not be built", check your tunnel group name to ensure that the tunnel-group name does not contain the following special characters: &, ", or <. The tunnel group name is added using the **tunnel-group webvpn-attributes** command.

## Related Commands

| Command | Description |
|---|---|
| authentication | Sets the authentication type for a tunnel group, such as saml. |
| idp | Sets the Idp for a third-party SAML identity provider. |

# sast

To specify the number of SAST certificates to create in the CTL record, use the **sast** command in ctl-file configuration mode. To set the number of SAST certificates in the CTL file back to the default value of 2, use the **no** form of this command.

**sast** *number_sasts*
**no sast** *number_sasts*

## Syntax Description

| | |
|---|---|
| *number_sasts* | Specifies the number of SAST keys to create. The default is 2. The maximum allowed is 5. |

## 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 |
| Ctl-file configuration | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(4) | The command was added. |

## Usage Guidelines

CTL files are signed by a System Administrator Security Token (SAST).

Because the Phone Proxy generates the CTL file, it needs to create the SAST key to sign the CTL file itself. This key can be generated on the ASA. A SAST is created as a self-signed certificate.

Typically, a CTL file contains more than one SAST. In case a SAST is not recoverable, the other one can be used to sign the file later.

## Examples

The following example shows the use of the **sast** command to create 5 SAST certificates in the CTL file:

```
ciscoasa
(
config-ctl-file
)# sast 5
```

## Related Commands

| Command | Description |
|---|---|
| **ctl-file (global)** | Specifies the CTL file to create for Phone Proxy configuration or the CTL file to parse from Flash memory. |

| Command | Description |
|---|---|
| **ctl-file (phone-proxy)** | Specifies the CTL file to use for Phone Proxy configuration. |
| **phone-proxy** | Configures the Phone Proxy instance. |

# scansafe

To enable Cloud Web Security inspection for a context, use the **scansafe** command in context configuration mode. To disable Cloud Web Security, use the **no** form of this command.

**scansafe** [ **license** *key* ]
**no scansafe** [ **license** *key* ]

## Syntax Description

| | |
|---|---|
| **license** *key* | Enters an authentication key for this context. If you do not specify a key, the context uses the license configured in the system configuration. The ASA sends the authentication key to the Cloud Web Security proxy servers to indicate from which organization the request comes. The authentication key is a 16-byte hexadecimal number. |

## Command Default

By default, the context uses the license entered in the system configuration.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.0(1) | This command was added. |

## Usage Guidelines

In multiple context mode, you must allow Cloud Web Security per context.

## Examples

The following sample configuration enables Cloud Web Security in context one with the default license and in context two with the license key override:

```
! System Context
!
scansafe general-options
server primary ip 180.24.0.62 port 8080
retry-count 5
license 366C1D3F5CE67D33D3E9ACEC265261E5
!
context one
 allocate-interface GigabitEthernet0/0.1
 allocate-interface GigabitEthernet0/1.1
 allocate-interface GigabitEthernet0/3.1
 scansafe
 config-url disk0:/one_ctx.cfg
!
context two
 allocate-interface GigabitEthernet0/0.2
```

```
 allocate-interface GigabitEthernet0/1.2
 allocate-interface GigabitEthernet0/3.2
 scansafe license 366C1D3F5CE67D33D3E9ACEC26789534
 config-url disk0:/two_ctx.cfg
!
```

## Related Commands

| Command | Description |
|---|---|
| **class-map type inspect scansafe** | Creates an inspection class map for whitelisted users and groups. |
| **default user group** | Specifies the default username and/or group if the ASA cannot determine the identity of the user coming into the ASA. |
| **http** [ **s** ] (parameters) | Specifies the service type for the inspection policy map, either HTTP or HTTPS. |
| **inspect scansafe** | Enables Cloud Web Security inspection on the traffic in a class. |
| **license** | Configures the authentication key that the ASA sends to the Cloud Web Security proxy servers to indicate from which organization the request comes. |
| **match user group** | Matches a user or group for a whitelist. |
| **policy-map type inspect scansafe** | Creates an inspection policy map so you can configure essential parameters for the rule and also optionally identify the whitelist. |
| **retry-count** | Enters the retry counter value, which is the amount of time that the ASA waits before polling the Cloud Web Security proxy server to check its availability. |
| **scansafe** | In multiple context mode, allows Cloud Web Security per context. |
| **scansafe general-options** | Configures general Cloud Web Security server options. |
| **server** { **primary** \| **backup** } | Configures the fully qualified domain name or IP address of the primary or backup Cloud Web Security proxy servers. |
| **show conn scansafe** | Shows all Cloud Web Security connections, as noted by the capitol Z flag. |
| **show scansafe server** | Shows the status of the server, whether it's the current active server, the backup server, or unreachable. |
| **show scansafe statistics** | Shows total and current http connections. |
| **user-identity monitor** | Downloads the specified user or group information from the AD agent. |
| **whitelist** | Performs the whitelist action on the class of traffic. |

# scansafe general-options

To configure communication with the Cloud Web Security proxy server, use the **scansafe general-options** command in global configuration mode. To remove the server configuration, use the **no** form of this command.

**scansafe general-options**
**no scansafe general-options**

## 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 |
|---|---|
| 9.0(1) | This command was added. |

## Usage Guidelines

You can configure a primary and backup proxy server for Cloud Web Security.

## Examples

The following example configures a primary server:

```
scansafe general-options
 server primary ip 180.24.0.62 port 8080
 retry-count 5
 license 366C1D3F5CE67D33D3E9ACEC265261E5
```

## Related Commands

| Command | Description |
|---|---|
| **class-map type inspect scansafe** | Creates an inspection class map for whitelisted users and groups. |
| **default user group** | Specifies the default username and/or group if the ASA cannot determine the identity of the user coming into the ASA. |
| **health-check application** | Enables Cloud Web Security application health checking for failover. |
| **http**[**s**] (parameters) | Specifies the service type for the inspection policy map, either HTTP or HTTPS. |
| **inspect scansafe** | Enables Cloud Web Security inspection on the traffic in a class. |

| Command | Description |
|---|---|
| **license** | Configures the authentication key that the ASA sends to the Cloud Web Security proxy servers to indicate from which organization the request comes. |
| **match user group** | Matches a user or group for a whitelist. |
| **policy-map type inspect scansafe** | Creates an inspection policy map so you can configure essential parameters for the rule and also optionally identify the whitelist. |
| **retry-count** | Enters the retry counter value, which is the amount of time that the ASA waits before polling the Cloud Web Security proxy server to check its availability. |
| **scansafe** | In multiple context mode, allows Cloud Web Security per context. |
| **scansafe general-options** | Configures general Cloud Web Security server options. |
| **server** {**primary** \| **backup**} | Configures the fully qualified domain name or IP address of the primary or backup Cloud Web Security proxy servers. |
| **show conn scansafe** | Shows all Cloud Web Security connections, as noted by the capitol Z flag. |
| **show scansafe server** | Shows the status of the server, whether it's the current active server, the backup server, or unreachable. |
| **show scansafe statistics** | Shows total and current http connections. |
| **user-identity monitor** | Downloads the specified user or group information from the AD agent. |
| **whitelist** | Performs the whitelist action on the class of traffic. |

# scep-enrollment enable

To enable or disable the Simple Certificate Enrollment Protocol for a tunnel group, use the **scep-enrollment enable** command in tunnel-group general-attributes mode.

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

**scep-enrollment enable**
**no scep-enrollment enable**

## Syntax Description

This command has no arguments or keywords.

## Command Default

By default, this command is not present in the tunnel group configuration.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 8.4(1) | This command was added. |

## Usage Guidelines

Only the Cisco Secure Client, Release 3.0 and later, supports this feature.

The ASA can proxy SCEP requests between Secure Client and a third-party certificate authority. The certificate authority only needs to be accessible to the ASA if it is acting as the proxy. For the ASA to provide this service, the user must authenticate using any of the methods supported by AAA before the ASA sends an enrollment request. You can also use Host Scan and dynamic access policies to enforce rules of eligibility to enroll.

The ASA supports this feature only with an AnyConnect SSL or IKEv2 VPN session. It supports all SCEP-compliant certificate authorities, including IOS CS, Windows Server 2003 CA, and Windows Server 2008 CA.

Clientless (browser-based) access does not support SCEP Proxy, although WebLaunch—clientless-initiated Secure Client—does support it.

The ASA does not support polling for certificates.

The ASA supports load balancing for this feature.

## Examples

The following example, entered in global configuration mode, creates a remote access tunnel group named remotegrp and enables SCEP for the group policy:

```
ciscoasa(config)# tunnel-group remotegrp type remote-access
ciscoasa(config)# tunnel-group remotegrp general-attributes
ciscoasa(config-tunnel-general)# scep-enrollment enable
INFO: 'authentication aaa certificate' must be configured to complete setup of this option.
```

## Related Commands

| Command | Description |
|---|---|
| crypto ikev2 enable | Enables IKEv2 negotiation on the interface on which IPsec peers communicate. |
| scep-forwarding-url | Enrolls the SCEP certificate authority for the group policy. |
| secondary-pre-fill-username clientless | Supplies a common, secondary password when a certificate is unavailable for WebLaunch support of the SCEP proxy. |
| **secondary-authentication-server-group** | Supplies the username when a certificate is unavailable. |

# scep-forwarding-url

To enroll an SCEP certificate authority for a group policy, use the **scep-forwarding-url** command in group-policy configuration mode.

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

**scep-forwarding-url** { **none | value** [ *URL* ] }
**no scep-forwarding-url**

## Syntax Description

| | |
|---|---|
| **none** | Specifies no certificate authority for the group policy. |
| *URL* | Specifies the SCEP URL of the certificate authority. |
| **value** | Enables this feature for clientless connections. |

## Command Default

By default, this command is not present.

## 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 |
|---|---|
| 8.4(1) | This command was added. |

## Usage Guidelines

Enter this command once per group policy to support a third-party digital certificate.

## Examples

The following example, entered in global configuration mode, creates a group policy named FirstGroup and enrolls a certificate authority for the group policy:

```
ciscoasa(config)# group-policy FirstGroup internal
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# scep-forwarding-url value http://ca.example.com:80/
Attempting to retrieve the CA/RA certificate(s) using the URL. Please wait ...
```

## Related Commands

| Command | Description |
|---|---|
| crypto ikev2 enable | Enables IKEv2 negotiation on the interface on which IPsec peers communicate. |
| scep-enrollment enable | Enables Simple Certificate Enrollment Protocol for a tunnel group. |

| Command | Description |
|---|---|
| secondary-pre-fill-username clientless | Supplies a common, secondary password when a certificate is unavailable for WebLaunch support of the SCEP proxy. |
| **secondary-authentication-server-group** | Supplies the username when a certificate is unavailable. |

# secondary

To set the preferred unit for a failover group when using the **preempt** command, use the **secondary** command in failover group configuration mode. To restore the default value, use the **no** form of this command.

**secondary**
**no secondary**

## Syntax Description

This command has no arguments or keywords.

## Command Default

If **primary** or **secondary** is not specified for a failover group, the failover group defaults to **primary**.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 9.0(1) | Earlier software versions allowed “simultaneous” boot up so that the failover groups did not require the **preempt** command to become active on the preferred unit. However, this functionality has now changed so that both failover groups become active on the first unit to boot up. |

## Usage Guidelines

Assigning a **primary** or **secondary** preference to a failover group specifies which unit the failover group becomes active on when you set the **preempt** command. Both failover groups become active on the first unit that boots up (even if it seems like they boot simultaneously, one unit becomes active first), despite the **primary** or **secondary** setting for the group. When the other unit comes online, any failover groups that have the second unit as a priority do not become active on the second unit unless the failover group is configured with the **preempt** command or is manually forced to the other unit with the **no failover active** command. If the failover group is configured with the **preempt** command, the failover group automatically becomes active on the designated unit.

## Examples

The following example configures failover group 1 with the primary unit as the higher priority and failover group 2 with the secondary unit as the higher priority. Both failover groups are configured with the **preempt** command, so the groups will automatically become active on their preferred unit as the units become available.

```
ciscoasa(config)# failover group 1

ciscoasa(config-fover-group)# primary
ciscoasa(config-fover-group)# preempt 100
ciscoasa(config-fover-group)# exit
```

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

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

## Related Commands

| Command | Description |
|---|---|
| **failover group** | Defines a failover group for Active/Active failover. |
| **preempt** | Forces the failover group to become active on its preferred unit when the unit becomes available. |
| **primary** | Gives the primary unit a higher priority than the secondary unit. |

# secondary-authentication-server-group

To specify a secondary authentication server group to associate with the session when double authentication is enabled, use the **secondary-authentication-server-group** command in tunnel-group general-attributes mode. To remove the attribute from the configuration, use the **no** form of this command.

**secondary-authentication-server-group** [*interface_name*] {**none | LOCAL |***groupname* [ **LOCAL** ] } [ **use-primary-username** ]
**no secondary-authentication-server-group**

## Syntax Description

| | |
|---|---|
| *interface_name* | (Optional) Specifies the interface where the IPsec tunnel terminates. |
| **LOCAL** | (Optional) Requires authentication against the local user database if all of the servers in the server group have been deactivated due to communication failures. If the server group name is either **LOCAL** or **NONE,** do not use the **LOCAL** keyword here. |
| **none** | (Optional) Specifies the server group name as **NONE** , indicating that authentication is not required. |
| *groupname* [ **LOCAL** ] | Identifies the previously configured authentication server or group of servers. Optionally, this can be the LOCAL group. |
| **use-primary-username** | Use the primary username as the username for the secondary authentication. |

## Command Default

The default value 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 |
| Tunnel-group general-attributes configuration | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.2(1) | This command was added. |

## Usage Guidelines

This command is meaningful only when double authentication is enabled. The **secondary-authentication-server-group** command specifies the secondary AAA server group. The secondary server group cannot be an SDI server group.

If the use-primary-username keyword is configured, then only one username is requested in the login dialog.

If the usernames are extracted from a digital certificate, only the primary username is used for authentication.

## Examples

The following example, entered in global configuration mode, creates a remote access tunnel group named remotegrp and specifies the use of the group sdi_server as the primary server group and the group ldap_ server as the secondary authentication server group for the connection:

```
ciscoasa(config)# tunnel-group remotegrp type remote-access
ciscoasa(config)# tunnel-group remotegrp general-attributes
ciscoasa(config-tunnel-webvpn)# authentication-server-group sdi_server
ciscoasa(config-tunnel-webvpn)# secondary-authentication-server-group ldap_server
ciscoasa(config-tunnel-webvpn)#
```

## Related Commands

| Command | Description |
|---|---|
| pre-fill-username | Enables the pre-fill username feature. |
| show running-config tunnel-group | Shows the indicated tunnel-group configuration. |
| tunnel-group general-attributes | Specifies the general attributes for the named tunnel-group. |
| username-from-certificate | Specifies the field in a certificate to use as the username for authorization. |

# secondary-color

To set a secondary color for the WebVPN login, home page, and file access page, use the **secondary-color** command in webvpn configuration mode. To remove a color from the configuration and reset the default, use the **no** form of this command.

**secondary-color** [ *color* ]
**no secondary-color**

## Syntax Description

*color* (Optional) Specifies the color. You can use a comma separated RGB value, an HTML color value, or the name of the color if recognized in HTML.

- RGB format is 0,0,0, a range of decimal numbers from 0 to 255 for each color (red, green, blue); the comma separated entry indicates the level of intensity of each color to combine with the others.
- HTML format is #000000, six digits in hexadecimal format; the first and second represent red, the third and fourth green, and the fifth and sixth represent blue.
- Name length maximum is 32 characters

## Command Default

The default secondary color is HTML #CCCCFF, a lavender shade.

## 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 | — | — | • Yes |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |

## Usage Guidelines

The number of RGB values recommended for use is 216, many fewer than the mathematical possibilities. Many displays can handle only 256 colors, and 40 of those look differently on MACs and PCs. For best results, check published RGB tables. To find RGB tables online, enter RGB in a search engine.

## Examples

The following example shows how to set an HTML color value of #5F9EAO, which is a teal shade:

```
ciscoasa
(config)#
 webvpn
ciscoasa(config-webvpn)# secondary-color #5F9EAO
```

**Related Commands**

| Command | Description |
|---|---|
| **title-color** | Sets a color for the WebVPN title bar on the login, home page, and file access page |

# secondary-pre-fill-username

To enable the extraction of a username from a client certificate for use in double authentication for a clientless or an Secure Client connection, use the **secondary-pre-fill-username** command in tunnel-group webvpn-attributes mode. To remove the attribute from the configuration, use the **no** form of this command.

**secondary-pre-fill-username** { **clientless | ssl-client** } [ **hide** ]
**secondary-pre-fill-username** { **clientless | ssl-client** } **hide** [ **use-primary-password |**
**use-common-password** [ *type_num* ] *password* ]
**no secondary-pre-fill-username**

## Syntax Description

| | |
|---|---|
| **clientless** | Enables this feature for clientless connections. |
| **hide** | Hides the username to be used for authentication from the VPN user. |
| *password* | Enter the password string. |
| **client**<br>**ssl-client** | Enables this feature for AnyConnect VPN client connections. Use the **client** keyword in 9.8(1)+. |
| *type_num* | Enter one of the following options:<br>• 0 if the password to be entered is plain text.<br>• 8 if the password to be entered is encrypted. The password appears as asterisks as you type. |
| **use-common-password** | Specifies a common secondary authentication password to use without prompting the user for it. |
| **use-primary-password** | Reuses the primary authentication password for secondary authentication without prompting the user for it. |

## Command Default

This feature is disabled by default. Entering this command without the **hide** keyword reveals the extracted username to the VPN user. The user receives a password prompt if you specify neither the use-primary-password nor the use-common-password keywords. The default value of *type_num* is 8.

## 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 webvpn-attributes configuration | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.2(1) | This command was added. |
| 8.3(2) | The [ **use-primary-password** \| **use-common-password** [*type_num* ] *password* ] option was added. |
| 9.8(1) | The **ssl-client** keyword was changed to **client**. |

## Usage Guidelines

To enable this feature, you must also enter the **secondary-username-from-certificate** command in tunnel-group general-attributes mode.

This command is meaningful only if double authentication is enabled. The **secondary-pre-fill-username** command enables the use of a username extracted from the certificate field specified in the **secondary-username-from-certificate** command as the username for secondary username/password authentication. To use this secondary-pre-fill username-from-certificate feature, you must configure both commands.

**Note** Clientless and SSL-client connections are not mutually exclusive options. Only one can be specified per command line, but both can be enabled at the same time.

If you hide the second username and use a primary or common password, the user experience is similar to single authentication. Using the primary or common password makes the use of device certificates to authenticate a device a seamless user experience.

The **use-primary-password** keyword specifies the use of the primary password as the secondary password for all authentications.

The **use-common-password** keyword specifies the use of a common secondary password for all secondary authentications. If a device certificate installed on the endpoint contains a BIOS ID or some other identifier, a secondary authentication request can use the pre-filled BIOS ID as the second username and use a common password configured for all authentications in that tunnel group.

## Examples

The following example creates an IPsec remote access tunnel group named remotegrp, and specifies the reuse of a name from the digital certificate on the endpoint as the name to be used for an authentication or authorization query when the connections are browser-based.

```
ciscoasa(config)# tunnel-group remotegrp type ipsec_ra
ciscoasa(config)# tunnel-group remotegrp webvpn-attributes
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username clientless
```

The following example performs the same function as the previous command, but hides the extracted username from the user:

```
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username clientless hide
```

The following example performs the same function as the previous command, except that it applies only to Secure Client connections:

```
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username client hide
```

The following example hides the username and reuses the primary authentication password for secondary authentication without prompting the user:

```
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username client hide use-primary-password
```

The following example hides the username and uses the password you enter for secondary authentication:

```
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username client hide use-common-password
 **********
```

## Related Commands

| Command | Description |
|---|---|
| **pre-fill-username** | Enables the pre-fill username feature. |
| **show running-config tunnel-group** | Shows the indicated tunnel-group configuration. |
| **tunnel-group general-attributes** | Specifies the general attributes for the named tunnel-group. |
| **username-from-certificate** | Specifies the field in a certificate to use as the username for authorization. |

# secondary-text-color

To set the secondary text color for the WebVPN login, home page and file access page, use the **secondary-text-color** command in webvpn mode. To remove the color from the configuration and reset the default, use the **no** form of this command.

**secondary-text-color** [ *black* | *white* ]
**no secondary-text-color**

## Syntax Description

| | |
|---|---|
| auto | Chooses black or white based on the settings for the text-color command. That is, if the primary color is black, this value is white. |
| black | The default secondary text color is black. |
| white | You can change the text color to white. |

## Command Default

The default secondary text color is black.

## 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 | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |

## Examples

The following example shows how to set the secondary text color to white:

```
ciscoasa
(config)#
 webvpn
ciscoasa(config-webvpn)# secondary-text-color white
```

## Related Commands

| Command | Description |
|---|---|
| text-color | Sets a color for text in the WebVPN title bar on the login, home page and file access page |

# secondary-username -from-certificate

To specify the field in a certificate to use as the secondary username for double authentication for a clientless or AnyConnect (SSL-client) connection, use the **secondary-username-from-certificate** command in tunnel-group general-attributes mode.

To remove the attribute from the configuration and restore default values, use the **no** form of this command.

**secondary-username-from-certificate** { *primary-attr* [ *secondary-attr* ] | **use-entire-name** | **use-script** }
**no secondary-username-from-certificate**

## Syntax Description

| | |
|---|---|
| *primary-attr* | Specifies the attribute to use to derive a username for an authorization query from a certificate. If pre-fill-username is enabled, the derived name can also be used in an authentication query. |
| *secondary-attr* | (Optional) Specifies an additional attribute to use with the primary attribute to derive a username for an authentication or authorization query from a digital certificate. If pre-fill-username is enable, the derived name can also be used in an authentication query. |
| **use-entire-name** | Specifies that the ASA must use the entire subject DN (RFC1779) to derive a name for an authorization query from a digital certificate. |
| use-script | Specifies the use of a script file generated by Adaptive Security Device Manager (ASDM) to extract the DN fields from a certificate for use as a username. |

## Command Default

This feature is disabled by default and is meaningful only when double authentication 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 |
| Tunnel-group general-attributes configuration | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 8.2(1) | This command was added. |

## Usage Guidelines

This command is meaningful only when double authentication is enabled.

When double authentication is enabled. this command selects one or more fields in a certificate to use as the username. The **secondary-username-from-certificate** command forces the security appliance to use the specified certificate field as the second username for the second username/password authentication.

To use this derived username in the pre-fill username from certificate feature for the secondary username/password authentication or authorization, you must also configure the **pre-fill-username** and **secondary-pre-fill-username** commands in tunnel-group webvpn-attributes mode. That is, to use the secondary pre-fill username feature, you must configure both commands.

Possible values for primary and secondary attributes include the following:

| Attribute | Definition |
|---|---|
| C | Country: the two-letter country abbreviation. These codes conform to ISO 3166 country abbreviations. |
| CN | Common Name: the name of a person, system, or other entity. Not available a s a secondary attribute. |
| DNQ | Domain Name Qualifier. |
| EA | E-mail address. |
| GENQ | Generational Qualifier. |
| GN | Given Name. |
| I | Initials. |
| L | Locality: the city or town where the organization is located. |
| N | Name. |
| O | Organization: the name of the company, institution, agency, association or other entity. |
| OU | Organizational Unit: the subgroup within the organization (O). |
| SER | Serial Number. |
| SN | Surname. |
| SP | State/Province: the state or province where the organization is located |
| T | Title. |
| UID | User Identifier. |
| UPN | User Principal Name. |
| use-entire-name | Use entire DN name. Not available a s a secondary attribute. |
| use-script | Use a script file generated by ASDM. |

**Note** If you also specify the **secondary-authentication-server-group** command, along with the **secondary-username-from-certificate command, only** the primary username is used for authentication.

## Examples

The following example, entered in global configuration mode, creates a remote access tunnel group named remotegrp and specifies the use of CN (Common Name) as the primary attribute and OU as the secondary attribute to use to derive a name for an authorization query from a digital certificate:

```
ciscoasa(config)# tunnel-group remotegrp type remote-access
ciscoasa(config)# tunnel-group remotegrp general-attributes
ciscoasa(config-tunnel-general)# username-from-certificate CN
ciscoasa(config-tunnel-general)# secondary-username-from-certificate OU
ciscoasa(config-tunnel-general)#
```

The following example shows how to modify the tunnel-group attributes to configure the pre-fill username.

```
username-from-certificate {use-entire-name | use-script | <primary-attr>} [secondary-attr]
 secondary-username-from-certificate {use-entire-name | use-script | <primary-attr>}
[secondary-attr] ; used only for double-authentication
```

## Related Commands

| Command | Description |
|---|---|
| pre-fill-username | Enables the pre-fill username feature. |
| secondary-pre-fill-username | Enables username extraction for clientless or Secure Client connection |
| username-from-certificate | Specifies the field in a certificate to use as the username for authorization. |
| show running-config tunnel-group | Shows the indicated tunnel-group configuration. |
| **secondary-authentication-server-group** | Specifies the secondary AAA server group. If the usernames are extracted from a digital certificate, only the primary username is used for authentication. |

# secondary-username-from-certificate-choice

To select the certificate from where the username should be used for pre-fill username field for secondary authentication or authorization, use the **secondary-username-from-certificate-choice** command. Use this command in tunnel-group general-attributes mode. To use the username from the default certificate, use the **no** form of the command.

**secondary-username-from-certificate-choice** { **first-certificate | second-certificate** }
**no secondary-username-from-certificate-choice** { **first-certificate | second-certificate** }

## Syntax Description

| | |
|---|---|
| **first-certificate** | Specifies if the username from the machine certificate sent in SSL or IKE to be used in pre-fill username field for secondary authentication. |
| **second-certificate** | Specifies if the username from the user certificate from client to be used in pre-fill username field for secondary authentication. |

## Command Default

The username for prefill is retrieved from the second certificate 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 | — |

## Command History

| Release | Modification |
|---|---|
| 9.14(1) | This command was added. |

## Usage Guidelines

The multiple certificates option allows certificate authentication of both the machine and user via certificates. The pre-fill username field allows a field from the certificates to be parsed and used for subsequent (primary and secondary)AAA authentication in a AAA and certificate authenticated connection. The username for prefill is always retrieved from the second (user) certificate received from the client.

Beginning with 9.14(1), ASA allows you to choose whether the first certificate (machine certificate) or second certificate (user certificate) should be used to derive the username for the pre-fill username field.

This command is available and can be configured for any tunnel groups irrespective of the authentication type (aaa, certificate, or multiple-certificate). However, the configuration takes effect only for Multiple Certificate Authentication (multiple-certificate or aaa multiple-certificate). When the option is not used for Multiple Certificate Authentication, the second certificate is used by default for authentication or authorization purpose.

## Examples

The following example shows how to configure the certificate to be used for prefill username for primary and secondary authentication or authorization:

```
ciscoasa(config)#tunnel-group tg1 type remote-access
ciscoasa(config)#tunnel-group tg1 general-attributes
ciscoasa(config-tunnel-general)# address-pool IPv4
ciscoasa(config-tunnel-general)# secondary-authentication-server-group LOCAL/<Auth-Server>

ciscoasa(config-tunnel-general)# username-from-certificate-choice first-certificate
ciscoasa(config-tunnel-general)# secondary-username-from-certificate-choice first-certificate
ciscoasa(config)# tunnel-group tg1 webvpn-attributes
ciscoasa(config-tunnel-webvpn)# authentication aaa multiple-certificate
ciscoasa(config-tunnel-webvpn)# pre-fill-username client
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username client
```

## Related Commands

| Command | Description |
|---|---|
| **username-from-certificate-choice** | Specify the certificate option for primary authentication. |

# secure-unit-authentication

To enable secure unit authentication, use the **secure-unit-authentication enable** command in group-policy configuration mode. To disable secure unit authentication, use the **secure-unit-authentication disable** command. To remove the secure unit authentication attribute from the running configuration, use the **no** form of this command.

**secure-unit-authentication** { **enable | disable** }
**no secure-unit-authentication**

## Syntax Description

| | |
|---|---|
| **disable** | Disables secure unit authentication. |
| **enable** | Enables secure unit authentication. |

## Command Default

Secure unit authentication 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

Secure unit authentication requires that you have an authentication server group configured for the tunnel group the hardware client(s) use.

If you require secure unit authentication on the primary ASA, be sure to configure it on any backup servers as well.

The **no** option allows inheritance of a value for secure unit authentication from another group policy.

Secure unit authentication provides additional security by requiring VPN hardware clients to authenticate with a username and password each time the client initiates a tunnel. With this feature enabled, the hardware client does not have a saved username and password.

**Note** With this feature enabled, to bring up a VPN tunnel, a user must be present to enter the username and password.

## Examples

The following example shows how to enable secure unit authentication for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# secure-unit-authentication
 enable
```

## Related Commands

| Command | Description |
|---|---|
| **ip-phone-bypass** | Lets IP phones connect without undergoing user authentication. Secure unit authentication remains in effect. |
| **leap-bypass** | Lets LEAP packets from wireless devices behind a VPN hardware client travel across a VPN tunnel prior to user authentication, when enabled. This lets workstations using Cisco wireless access point devices establish LEAP authentication. Then they authenticate again per user authentication. |
| **user-authentication** | Requires users behind a hardware client to identify themselves to the ASA before connecting. |

# security-group

To add a security group to a security object group for use with Cisco TrustSec, use the **security-group** command in object-group security configuration mode. To remove the security group, use the **no** form of this command.

**security-group** { **tag** *sgt##* | **name** *sg_name* }
**no security-group** { **tag** *sgt##* | **name** *sg_name* }

## Syntax Description

| | |
|---|---|
| **tag** *sgt#* | Specifies the security group object as an inline tag. Enter a number from 1 to 65533 for a Tag security type.<br><br>An SGT is assigned to a device through IEEE 802.1X authentication, web authentication, or MAC authentication bypass (MAB) by the ISE. Security group names are created on the ISE and provide user-friendly names for security groups. The security group table maps SGTs to security group names. |
| **name** *sg_name* | Specifies the security group object as a named object. Enter a 32-byte case-sensitive string for a Name security type. The *sg_name* can contain any character including [a-z], [A-Z], [0-9], [!@#$%^&()-_{}. ]. |

## 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 |
| Object-group security configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.0(1) | This command was added. |

## Usage Guidelines

You can create security group object groups for use in features that support Cisco TrustSec by including the group in an extended ACL, which in turn can be used in an access rule, for example.

When integrated with Cisco TrustSec, the ASA downloads security group information from the ISE. The ISE acts as an identity repository, by providing Cisco TrustSec tag to user identity mapping and Cisco TrustSec tag to server resource mapping. You provision and manage security group access lists centrally on the ISE.

However, the ASA might have localized network resources that are not defined globally that require local security groups with localized security policies. Local security groups can contain nested security groups that are downloaded from the ISE. The ASA consolidates local and central security groups.

To create local security groups on the ASA, you create a local security object group. A local security object group can contain one or more nested security object groups or Security IDs or security group names. User can also create a new Security ID or security group name that does not exist on the ASA.

You can use the security object groups you create on the ASA to control access to network resources. You can use the security object group as part of an access group or service policy.

## Examples

The following example shows how to configure a security group object:

```
ciscoasa(config)# object-group security mktg-sg
ciscoasa(config)# security-group name mktg
ciscoasa(config)# security-group tag 1
```

The following example shows how to configure a security group object:

```
ciscoasa(config)# object-group security mktg-sg-all
ciscoasa(config)# security-group name mktg-managers
ciscoasa(config)# group-object mktg-sg // nested object-group
```

## Related Commands

| Command | Description |
|---|---|
| **object-group security** | Creates a security group object. |

# security-group-tag

To configure a security group tag attribute in a remote access VPN group policy or for a user in the LOCAL user database, use the **security-group-tag value** command in group-policy or username configuration mode. To remove the security group tag attribute, use the **no** form of this command.

**security-group-tag** { **none | value** *sgt* }
**no security-group-tag** { **none | value** *sgt* }

## Syntax Description

| | |
|---|---|
| **none** | Do not set a security group tag for this group policy or user. |
| **value** *sgt* | Specifies the security group tag number. |

## Command Default

The default is **security-group-tag none**, which means that there is no security group tag in this attribute set.

## 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 or username configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.3(1) | This command was added. |

## Usage Guidelines

ASA supports security group tagging of VPN sessions. You can assign a Security Group Tag (SGT) to a VPN session using an external AAA server, or by configuring a security group tag for a local user or for a VPN group policy. This tag can then be propagated through the Cisco TrustSec system over Layer 2 Ethernet. Security group tags are useful on group policies and for local users when the AAA server cannot provide an SGT.

Following is the typical process for assigning an SGT to a VPN user:

**1.** A user connects to a remote access VPN that uses a AAA server group containing ISE servers.

**2.** The ASA requests AAA information from ISE, which might include an SGT. The ASA also assigns an IP address for the user's tunneled traffic.

**3.** The ASA uses AAA information to authenticate the user and creates a tunnel.

**4.** The ASA uses the SGT from AAA information and the assigned IP address to add an SGT in the Layer 2 header.

**5.** Packets that include the SGT are passed to the next peer device in the Cisco TrustSec network.

If there is no SGT in the attributes from the AAA server to assign to a VPN user, then the ASA uses the SGT in the group policy. If there is no SGT in the group policy, then tag 0x0 is assigned.

## Examples

The following example shows how to configure SGT attributes for a group policy.

```
ciscoasa(config-group-policy)# security-group-tag value 101
```

## Related Commands

| Command | Description |
|---|---|
| **show asp table cts sgt-map** | Displays the IP address-security group table mapping entries from the IP address-security group table mapping database maintained in the datapath. |
| **show cts sgt-map** | Displays the IP address-security group table manager entries in the control path. |

# security-level

To set the security level of an interface, use the **security-level** command in interface configuration mode. To set the security level to the default, use the **no** form of this command. The security level protects higher security networks from lower security networks by imposing additional protection between the two.

**security-level** *number*
**no security-level**

## Syntax Description

| | |
|---|---|
| *number* | An integer between 0 (lowest) and 100 (highest). |

## Command Default

By default, the security level is 0.

If you name an interface "inside" and you do not set the security level explicitly, then the ASA sets the security level to 100 (see the **nameif** command). You can change this level if desired.

## 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 **nameif** command to an interface configuration mode command. |

## Usage Guidelines

The level controls the following behavior:

- Network access—By default, there is an implicit permit from a higher security interface to a lower security interface (outbound). Hosts on the higher security interface can access any host on a lower security interface. You can limit access by applying an access list to the interface.

For same security interfaces, there is an implicit permit for interfaces to access other interfaces on the same security level or lower.

- Inspection engines —Some inspection engines are dependent on the security level. For same security interfaces, inspection engines apply to traffic in either direction.
- NetBIOS inspection engine—Applied only for outbound connections.
- OraServ inspection engine—If a control connection for the OraServ port exists between a pair of hosts, then only an inbound data connection is permitted through the ASA.
- Filt ering—HTTP(S) and FTP filtering applies only for outbound connections (from a higher level to a lower level).

For same security interfaces, you can filter traffic in either direction.

- NAT control—When you enable NAT control, you must configure NAT for hosts on a higher security interface (inside) when they access hosts on a lower security interface (outside).

Without NAT control, or for same security interfaces, you can choose to use NAT between any interface, or you can choose not to use NAT. Keep in mind that configuring NAT for an outside interface might require a special keyword.

- **established** com mand—This command allows return connections from a lower security host to a higher security host if there is already an established connection from the higher level host to the lower level host.

For same security interfaces, you can configure **established** commands for both directions.

Normally, interfaces on the same security level cannot communicate. If you want interfaces on the same security level to communicate, see the **same-security-traffic** command. You might want to assign two interfaces to the same level and allow them to communicate if you want to create more than 101 communicating interfaces, or you want protection features to be applied equally for traffic between two interfaces; for example, you have two departments that are equally secure.

If you change the security level of an interface, and you do not want to wait for existing connections to time out before the new security information is used, you can clear the connections using the **clear local-host** command.

## Examples

The following example configures the security levels for two interfaces to be 100 and 0:

```
ciscoasa(config)# interface gigabitethernet0/0
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
ciscoasa(config-if)# interface gigabitethernet0/1
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 10.1.2.1 255.255.255.0
ciscoasa(config-if)# no shutdown
```

## Related Commands

| Command | Description |
|---|---|
| **clear local-host** | Resets all connections. |
| **interface** | Configures an interface and enters interface configuration mode. |
| **nameif** | Sets the interface name. |
| **vlan** | Assigns a VLAN ID to a subinterface. |

# segment-id

To specify the VXLAN ID for a VNI interface, use the **segment-id** command in interface configuration mode. To remove the ID, use the **no** form of this command.

**segment-id***id*
**no segment-id** *id*

## Syntax Description

| | |
|---|---|
| *id* | Sets the ID between 1 and 16777215. |

## Command Default

No default behavior or values.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

The segment ID is used for VXLAN tagging.

## Examples

The following example configures the VNI 1 interface and specifies a segment ID of 1000:

```
ciscoasa(config)# interface vni 1
ciscoasa(config-if)# segment-id 1000
ciscoasa(config-if)# vtep-nve 1
ciscoasa(config-if)# nameif vxlan1000
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0 standby 10.1.1.2
ciscoasa(config-if)# ipv6 address 2001:0DB8::BA98:0:3210/48
ciscoasa(config-if)# security-level 50
ciscoasa(config-if)# 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. |

| Command | Description |
|---|---|
| **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. |

# send response

To send a RADIUS Accounting-Response Start and Accounting-Response Stop message to the sender of the RADIUS Accounting-Request Start and Stop messages, use the **send response** command in radius-accounting parameter configuration mode, which is accessed by using the **inspect radius-accounting** command.

This option is disabled by default.

**send response**
**no send response**

## Syntax Description

This command has no arguments or keywords.

## 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 |
| Radius-accounting parameter configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 7.2(1) | This command was added. |

## Examples

The following example shows how to send a response with RADIUS accounting:

```
hostname(config)# policy-map type inspect radius-accounting ra
ciscoasa(config-pmap)# send response
ciscoasa(config-pmap-p)# send response
```

## Related Commands

| Commands | Description |
|---|---|
| **inspect radius-accounting** | Sets inspection for RADIUS accounting. |
| **parameters** | Sets parameters for an inspection policy map. |

# seq-past-window

To set the action for packets that have past-window sequence numbers (the sequence number of a received TCP packet is greater than the right edge of the TCP receiving window), use the **seq-past-window** command in tcp-map configuration mode. To set the value back to the default, use the **no** form of this command. This command is part of the TCP normalization policy enabled using the **set connection advanced-options** command.

**seq-past-window** { **allow | drop** }
**no seq-past-window**

## Syntax Description

| | |
|---|---|
| **allow** | Allows packets that have past-window sequence numbers. This action is only allowed if the **queue-limit** command is set to 0 (disabled). |
| **drop** | Drops packets that have past-window sequence numbers. |

## Command Default

The default action is to drop packets that have past-window sequence numbers.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.2(4)/8.0(4) | This command was added. |

## Usage Guidelines

To enable TCP normalization, use the Modular Policy Framework:

**1.tcp-map**—Identifies the TCP normalization actions.

**a.seq-past-window**—In tcp-map configuration mode, you can enter the **seq-past-window** command and many others.

**2.class-map**—Identify the traffic on which you want to perform TCP normalization.

**3.policy-map**—Identify the actions associated with each class map.

**a.class**—Identify the class map on which you want to perform actions.

**b.set connection advanced-options**—Identify the tcp-map you created.

**4.service-policy**—Assigns the policy map to an interface or globally.

## Examples

The following example sets the ASA to allow packets that have past-window sequence numbers:

```
ciscoasa(config)# tcp-map tmap
ciscoasa(config-tcp-map)# seq-past-window allow
ciscoasa(config)# class-map cmap
ciscoasa(config-cmap)# match any
ciscoasa(config)# policy-map pmap
ciscoasa(config-pmap)# class cmap
ciscoasa(config-pmap)# set connection advanced-options tmap
ciscoasa(config)# service-policy pmap global
ciscoasa(config)#
```

## Related Commands

| Command | Description |
|---|---|
| **class-map** | Identifies traffic for a service policy. |
| **policy-map** | dentifies actions to apply to traffic in a service policy. |
| **queue-limit** | Sets the out-of-order packet limit. |
| **set connection advanced-options** | Enables TCP normalization. |
| service-policy | Applies a service policy to interface(s). |
| show running-config tcp-map | Shows the TCP map configuration. |
| **tcp-map** | Creates a TCP map and allows access to tcp-map configuration mode. |

# serial-number

To include the ASA serial number in the certificate during enrollment, use the **serial-number** command in crypto ca trustpoint configuration mode. To restore the default setting, use the **no** form of the command.

**serial-number**
**no serial-number**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The default setting is to not include the serial number.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |

## Examples

The following example enters crypto ca trustpoint configuration mode for trustpoint central, and includes the ASA serial number in the enrollment request for trustpoint central:

```
ciscoasa(config)# crypto ca trustpoint central
ciscoasa(ca-trustpoint)# serial-number
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ca trustpoint** | Enters trustpoint configuration mode. |

# server (pop3s, imap4s, smtps) (Deprecated)

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

To specify a default e-mail proxy server, use the **server** command in the applicable e-mail proxy configuration mode. To remove the attribute from the configuration, use the **no** version of this command. The ASA sends requests to the default e-mail server when the user connects to the e-mail proxy without specifying a server. If you do not configure a default server, and a user does not specify a server, the ASA returns an error.

**server** { *ipaddr or hostname* }
**no server**

## Syntax Description

| | |
|---|---|
| *hostname* | The DNS name of the default e-mail proxy server. |
| *ipaddr* | The IP address of the default e-mail proxy server. |

## Command Default

There is no default e-mail proxy server 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 |
| Pop3s configuration | • Yes | • Yes | — | — | • Yes |
| Imap4s configuration | • Yes | • Yes | — | — | • Yes |
| Smtps configuration | • Yes | • Yes | — | — | • Yes |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 9.5.2 | This command was deprecated. |

## Examples

The following example shows how to set a default POP3S e-mail server with an IP address. of 10.1.1.7:

```
ciscoasa
(config)#
 pop3s
```

```
ciscoasa(config-pop3s)# server 10.1.1.7
```

# server (scansafe general-options)

To configure the primary and backup Cloud Web Security proxy servers, use the **server** command in scansafe general-options configuration mode. To remove the server, use the **no** form of this command.

**server** { **primary | backup** } { **ip** *ip_address* | **fqdn** *fqdn* } [ **port** *port* ]
**no server** { **primary | backup** } { **ip** *ip_address* | **fqdn** *fqdn* } [ **port** *port*

## Syntax Description

| | |
|---|---|
| **backup** | Specifies that you are identifying the backup server. |
| **ip** *ip_address* | Specifies the server IP address. |
| **fqdn** *fqdn* | Specifies the server fully-qualified domain name (FQDN). |
| **port** *port* | (Optional) By default, the Cloud Web Security proxy server uses port 8080 for both HTTP and HTTPS traffic; do not change this value unless directed to do so. |
| **primary** | Specifies that you are identifying the primary server. |

## Command Default

The default port is 8080.

## 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 |
| Scansafe general-options configuration | • Yes | • Yes | • Yes | — | • Yes |

## Command History

| Release | Modification |
|---|---|
| 9.0(1) | This command was added. |

## Usage Guidelines

When you subscribe to the Cisco Cloud Web Security service, you are assigned a primary Cloud Web Security proxy server and backup proxy server. These servers are routinely polled to check for their availability. If your ASA is unable to reach the Cloud Web Security proxy server (for example, if no SYN/ACK packets arrive from the proxy server), then the proxy server is polled through a TCP three-way handshake to check its availability. If the proxy server is unavailable after a configured number of retries (the default is five), the server is declared as unreachable, and the backup proxy server becomes active.

**Note** You can further refine failover by checking the health of the Cloud Web Security application. In some cases, the server can complete the TCP three-way handshake, yet the Cloud Web Security application on the server is not functioning correctly. If you enable application health checking, the system can fail over to the backup server even if the three-way handshake completes, if the application itself does not respond. This provides a more reliable failover setup. Use the **health-check application** command to enable this extra check.

The ASA automatically falls back to the primary Cloud Web Security proxy server from the backup server after continued polling shows that the primary server is active for two consecutive retry count periods. You can change this polling interval using the **retry-count** command.

| Traffic Conditions Under Which Proxy Server Is Not Reachable | Server Timeout Calculation | Connection Timeout Result |
|---|---|---|
| High traffic | Client half open connection timeout + ASA TCP connection timeout | (30 + 30) = 60 seconds |
| Single connection failure | Client half open connection timeout + ((retry threshold - 1) x (ASA TCP connection timeout)) | (30 + ((5-1) x (30)) = 150 seconds |
| Idle—No connections are passing | 15 minutes + ((retry threshold) x (ASA TCP connection timeout)) | 900 + (5 x (30) = 1050 seconds |

## Examples

The following example configures a primary and backup server. You must enter the command separately for the primary and backup server.

```
scansafe general-options
 server primary ip 10.24.0.62 port 8080
 server backup ip 10.10.0.7 port 8080
 health-check application
 retry-count 7
 license 366C1D3F5CE67D33D3E9ACEC265261E5
```

## Related Commands

| Command | Description |
|---|---|
| **class-map type inspect scansafe** | Creates an inspection class map for whitelisted users and groups. |
| **default user group** | Specifies the default username and/or group if the ASA cannot determine the identity of the user coming into the ASA. |
| **health-check application** | Enables Cloud Web Security application health checking for failover. |
| **http** [ **s** ] (parameters) | Specifies the service type for the inspection policy map, either HTTP or HTTPS. |
| **inspect scansafe** | Enables Cloud Web Security inspection on the traffic in a class. |
| **license** | Configures the authentication key that the ASA sends to the Cloud Web Security proxy servers to indicate from which organization the request comes. |

| Command | Description |
|---|---|
| **match user group** | Matches a user or group for a whitelist. |
| **policy-map type inspect scansafe** | Creates an inspection policy map so you can configure essential parameters for the rule and also optionally identify the whitelist. |
| **retry-count** | Enters the retry counter value, which is the amount of time that the ASA waits before polling the Cloud Web Security proxy server to check its availability. |
| **scansafe** | In multiple context mode, allows Cloud Web Security per context. |
| **scansafe general-options** | Configures general Cloud Web Security server options. |
| **server** { **primary** \| **backup** } | Configures the fully qualified domain name or IP address of the primary or backup Cloud Web Security proxy servers. |
| **show conn scansafe** | Shows all Cloud Web Security connections, as noted by the capitol Z flag. |
| **show scansafe server** | Shows the status of the server, whether it's the current active server, the backup server, or unreachable. |
| **show scansafe statistics** | Shows total and current HTTP(S) connections. |
| **user-identity monitor** | Downloads the specified user or group information from the AD agent. |
| **whitelist** | Performs the whitelist action on the class of traffic. |

# server (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 **server** command in ssh pubkey-chain configuration mode. To remove a server and its host key, use the **no** form of this command.

**server** *ip_address*
**no server** *ip_address*

## Syntax Description

| | |
|---|---|
| *ip_address* | Specifies the SSH server IP address. |

## 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 |
| Ssh pubkey-chain 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, 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. |
| **ssh pubkey-chain** | Manually adds or deletes servers and their keys from the ASA database. |
| **ssh stricthostkeycheck** | Enables SSH host key checking for the on-board Secure Copy (SCP) client. |

# server authenticate-client

To enable the ASA to authenticate the TLS client during TLS handshake, use the **server authenticate-client** command in tls-proxy configuration mode.

To bypass client authentication, use the **no** form of this command.

**server authenticate-client**
**no server authenticate-client**

## Syntax Description

This command has arguments or keywords.

## Command Default

This command is enabled by default, which means the TLS client is required to present a certificate during handshake with the ASA.

## 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 |
| Tls-proxy configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(4) | This command was added. |

## Usage Guidelines

Use the **server authenticate-client** command to control whether a client authentication is required during TLS Proxy handshake. When enabled (by default), the security appliance sends a Certificate Request TLS handshake message to the TLS client, and the TLS client is required to present its certificate.

Use the **no** form of this command to disable client authentication. Disabling TLS client authentication is suitable when the ASA must interoperate with CUMA client or clients such as a Web browser that are incapable of sending a client certificate.

## Examples

The following example configures a TLS proxy instance with client authentication disabled:

```
ciscoasa(config)# tls-proxy mmp_tls
ciscoasa(config-tlsp)# no server authenticate-client
ciscoasa(config-tlsp)# server trust-point cuma_server_proxy
```

## Related Commands

| Command | Description |
|---|---|
| **tls-proxy** | Configures the TLS proxy instance. |

# server cipher-suite

To define the ciphers that the TLS proxy server can use, use the **server cipher suite** command in tls-proxy configuration mode. To use the global cipher setting, use the **no** form of this command.

**server cipher-suite** *cipher_list*
**no server cipher-suite** *cipher_list*

## Syntax Description

*cipher_list* Sets the ciphers to include any combination of the following:

- **3des-sha1**
- **aes128-sha1**
- **aes256-sha1**
- **des-sha1**
- **null-sha1**
- **rc4-sha1**

Separate multiple options with spaces.

## Command Default

If you do not define the ciphers the TLS proxy can use, the proxy server uses the global cipher suite defined by the **ssl cipher** command. By default, the global cipher level is medium, which means all ciphers are available except for NULL-SHA, DES-CBC-SHA, and RC4-MD5.

## 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 |
| Tls-proxy configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.8(1) | We introduced this command. |

## Usage Guidelines

You can now set the SSL cipher suite when the ASA acts as a TLS proxy server. Formerly, you could only set global settings for the ASA using the **ssl cipher** command.

Specify the server cipher-suite command only if you want to use a different suite than the one generally available on the ASA (the **ssl cipher** command).

To set the minimum TLS version for all SSL server connections on the ASA, see the **ssl server-version** command. The default is TLS v1.0.

## Examples

The following example sets the TLS proxy server ciphers:

```
ciscoasa(config)# tls-proxy test
ciscoasa(config-tlsp)# server cipher-list aes128-sha1 aes256-sha1
```

## Related Commands

| Command | Description |
|---|---|
| **tls-proxy** | Defines a TLS proxy instance and sets the maximum number of sessions. |
| **client cipher-list** | Defines a TLS proxy client cipher suite. |

# server-port

To configure a AAA server port for a host, use the **server-port** command in aaa-server host mode. To remove the designated server port, use the **no** form of this command.

**server-port** *port-number*
**no server-port** *port-number*

## Syntax Description

| | |
|---|---|
| *port-number* | A port number in the range of 0 through 65535. |

## Command Default

The default server ports are as follows:

- SDI—5500
- LDAP—389
- Kerberos—88
- NT—139
- TACACS+—49

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |

## Examples

The following example configures an SDI AAA server named srvgrp1 to use server port number 8888:

```
ciscoasa
(config)#
aaa-server srvgrp1 protocol sdi
ciscoasa
(config-aaa-server-group)#
aaa-server srvgrp1 host 192.168.10.10
ciscoasa
(config-aaa-server-host)#
server-port 8888
```

**Related Commands**

| Command | Description |
|---|---|
| **aaa-server host** | Configures host-specific AAA server parameters. |
| **clear configure aaa-server** | Removes all AAA server configurations. |
| **show running-config aaa-server** | Displays AAA server statistics for all AAA servers, for a particular server group, for a particular server within a particular group, or for a particular protocol. |

# server-separator (pop3s, imap4s, smtps) (Deprecated)

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

To specify a character as a delimiter between the e-mail and VPN server names, use **server-separator** command in the applicable e-mail proxy mode. To revert to the default, ":", use the no form of this command.

**server-separator** { *symbol* }
**no server-separator**

## Syntax Description

| | |
|---|---|
| *symbol* | The character that separates the e-mail and VPN server names. Choices are "@," (at) "\|" (pipe), ":"(colon), "#" (hash), "," (comma), and ";" (semi-colon). |

## Command Default

The default is "@" (at).

## 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 |
| Pop3s | • Yes | — | • Yes | — | — |
| Imap4s | • Yes | — | • Yes | — | — |
| Smtps | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 9.5.2 | This command was deprecated. |

## Usage Guidelines

The server separator must be different from the name separator.

## Examples

The following example shows how to set a pipe (|) as the server separator for IMAP4S:

```
ciscoasa
(config)#
 imap4s
ciscoasa(config-imap4s)# server-separator |
```

**Related Commands**

| Command | Description |
|---|---|
| **name-separator** | Separates the e-mail and VPN usernames and passwords. |

# server trust-point

To specify the proxy trustpoint certificate to present during TLS handshake, use the **server trust-point** command in TLS server configuration mode.

**server trust-point** *proxy_trustpoint*

## Syntax Description

| | |
|---|---|
| *proxy_trustpoint* | Specifies the trustpoint defined by the **crypto ca trustpoint** command. |

## 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 |
| TLS-proxy configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(4) | This command was added. |

## Usage Guidelines

The trustpoint can be self-signed, enrolled with a certificate authority, or from an imported credential. The **server trust-point** command has precedence over the global **ssl trust-point** command.

The **server trust-point** command specifies the proxy trustpoint certificate presented during TLS handshake. The certificate must be owned by the ASA (identity certificate). The certificate can be self-signed, enrolled with a certificate authority, or from an imported credential.

Create TLS proxy instances for each entity that can initiate a connection. The entity that initiates the TLS connection is in the role of TLS client. Because the TLS Proxy has strict definition of client proxy and server proxy, two TLS proxy instances must be defined if either of the entities could initiate the connection.

**Note** When you are creating the TLS proxy instance to use with the Phone Proxy, the server trustpoint is the internal Phone Proxy trustpoint created the CTL file instance. The trustpoint name is in the form*internal_PP_<ctl-file_instance_name>*

## Examples

The following example shows the use of the **server trust-point** command to specify the proxy trustpoint certificate to present during TLS handshake:

```
ciscoasa
(config-tlsp)# server trust-point ent_y_proxy
```

**Related Commands**

| Command | Description |
|---|---|
| client (tls-proxy) | Configures trustpoints, keypairs, and cipher suites for a TLS proxy instance. |
| client trust-point | Specifies the proxy trustpoint certificate to present during TLS handshake. |
| **ssl trust-point** | Specifies the certificate trustpoint that represents the SSL certificate for an interface. |
| **tls-proxy** | Configures a TLS proxy instance. |

# server-type

To manually configure the LDAP server model, use the **server-type** command in aaa-server host configuration mode. The ASA supports the following server models:

- Microsoft Active Directory
- Sun Microsystems JAVA System Directory Server, formerly named the Sun ONE Directory Server
- Generic LDAP directory servers that comply with LDAPv3 (no password management)

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

**server-type** { **auto-detect | microsoft | sun | generic | openldap | novell** }
**no server-type** { **auto-detect | microsoft | sun | generic | openldap | novell** }

## Syntax Description

| | |
|---|---|
| auto-detect | Specifies that the ASA determines the LDAP server type through auto-detection. |
| generic | Specifies LDAP v3-compliant directory servers other than Sun and Microsoft LDAP directory servers. Password management is not supported with generic LDAP servers. |
| *microsoft* | Specifies that the LDAP server is a Microsoft Active Directory. |
| openldap | Specifies that the LDAP server is an OpenLDAP server. |
| novell | Specifies that the LDAP server is a Novell server. |
| sun | Specifies that the LDAP server is a Sun Microsystems JAVA System Directory Server. |

## Command Default

By default, auto-detection attempts to determine the server type.

## 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 | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 7.1(1) | This command was added. |
| 8.0(2) | Support for the OpenLDAP and Novell server types was added. |

## Usage Guidelines

The ASA supports LDAP version 3 and is compatible with the Sun Microsystems JAVA System Directory Server, the Microsoft Active Directory, and other LDAPv3 directory servers.

**Note** Sun—The DN configured on the ASA to access a Sun directory server must be able to access the default password policy on that server. We recommend using the directory administrator, or a user with directory administrator privileges, as the DN. Alternatively, you can place an ACI on the default password policy.

- Microsoft—You must configure LDAP over SSL to enable password management with Microsoft Active Directory.
- Generic—Password management features are not supported.

By default, the ASA auto-detects whether it is connected to a Microsoft directory server, a Sun LDAP directory server, or a generic LDAPv3 server. However, if auto-detection fails to determine the LDAP server type and if you know the server is either a Microsoft or Sun server, you can use the **server-type** command to manually configure the server as either a Microsoft or a Sun Microsystems LDAP server.

## Examples

The following example, entered in aaa-server host configuration mode, configures the server type for the LDAP server ldapsvr1 at IP address 10.10.0.1. The first example configures a Sun Microsystems LDAP server.

```
ciscoasa(config)# aaa-server ldapsvr1 protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server ldapsvr1 host 10.10.0.1
ciscoasa(config-aaa-server-host)# server-type sun
```

The following example specifies that the ASA use auto-detection to determine the server type:

```
ciscoasa(config)# aaa-server ldapsvr1 protocol LDAP
ciscoasa(config-aaa-server-group)# aaa-server ldapsvr1 host 10.10.0.1
ciscoasa(config-aaa-server-host)# server-type auto-detect
```

## Related Commands

| Command | Description |
|---|---|
| **ldap-over-ssl** | Specifies that SSL secures the LDAP client-server connection. |
| sasl-mechanism | Configures SASL authentication between the LDAP client and server. |
| **ldap attribute-map (global configuration mode)** | Creates and names an LDAP attribute map for mapping user-defined attribute names to Cisco LDAP attribute names. |

# service (ctl-provider)

To specify the port to which the Certificate Trust List provider listens, use the service command in CTL provider configuration mode. To remove the configuration, use the **no** form of this command.

**service port** *listening_port*
**no service port** *listening_port*

## Syntax Description

| | |
|---|---|
| **port** *listening_port* | Specifies the certificate to be exported to the client. |

## Command Default

Default port is 2444.

## 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 |
| Ctl provider configuration | • Yes | • Yes | • Tes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(2) | This command was added. |

## Usage Guidelines

Use the service command in CTL provider configuration mode to specify the port to which the CTL provider listens. The port must be the one listened to by the CallManager servers in the cluster (as configured under Enterprise Parameters on the CallManager administration page). The default port is 2444.

## Examples

The following example shows how to create a CTL provider instance:

```
ciscoasa(config)# ctl-provider my_ctl
ciscoasa(config-ctl-provider)# client interface inside 172.23.45.1
ciscoasa(config-ctl-provider)# client username CCMAdministrator password XXXXXX encrypted
ciscoasa(config-ctl-provider)# export certificate ccm_proxy
ciscoasa(config-ctl-provider)# ctl install
```

## Related Commands

| Commands | Description |
|---|---|
| client | Specifies clients allowed to connect to the CTL provider and also username and password for client authentication. |
| ctl | Parses the CTL file from the CTL client and install trustpoints. |
| ctl-provider | Configures a CTL provider instance in CTL provider mode. |

| Commands | Description |
|---|---|
| **export** | Specifies the certificate to be exported to the client |
| tls-proxy | Defines a TLS proxy instance and sets the maximum sessions. |

# service (global)

To enable resets for denied TCP connections, use the **service** command in global configuration mode. To disable resets, use the **no** form of this command.

**service** { **resetinbound** [ **interface** *interface_name* ] | **resetoutbound** [ **interface** *interface_name* ] | **resetoutside** }
**no service** { **resetinbound** [ **interface** *interface_name* ] | **resetoutbound** [ **interface** *interface_name* ] | **resetoutside** }

## Syntax Description

| | |
|---|---|
| **interface** *interface_name* | Enables or disables resets for the specified interface. |
| **resetinbound** | Sends TCP resets for all inbound TCP sessions that attempt to transit the ASA and are denied by the ASA based on access lists or AAA settings. The ASA also sends resets for packets that are allowed by an access list or AAA, but do not belong to an existing connection and are denied by the stateful firewall. Traffic between same security level interfaces is also affected. When this option is not enabled, the ASA silently discards denied packets. If you do not specify an interface, then this setting applies to all interfaces. |
| **resetoutbound** | Sends TCP resets for all outbound TCP sessions that attempt to transit the ASA and are denied by the ASA based on access lists or AAA settings. The ASA also sends resets for packets that are allowed by an access list or AAA, but do not belong to an existing connection and are denied by the stateful firewall. Traffic between same security level interfaces is also affected. When this option is not enabled, the ASA silently discards denied packets. This option is enabled by default. You might want to disable outbound resets to reduce the CPU load during traffic storms, for example. |
| **resetoutside** | Enables resets for TCP packets that terminate at the least secure interface and are denied by the ASA based on access lists or AAA settings. The ASA also sends resets for packets that are allowed by an access list or AAA, but do not belong to an existing connection and are denied by the stateful firewall. When this option is not enabled, the ASA silently discards the packets of denied packets.<br><br>We recommend that you use the resetoutside keyword with interface PAT. This keyword allows the ASA to terminate the IDENT from an external SMTP or FTP server. Actively resetting these connections avoids the 30-second timeout delay.<br><br>**Note** Connections are always reset for BGP and WebVPN (on the least secure interface) regardless of this option. |

## Command Default

By default, **service resetoutbound is** enabled for all interfaces.

## Command Modes

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

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

**Command History**

| Release | Modification |
|---|---|
| 7.1(1) | The **interface** keyword and the **resetoutbound** command were added. |

**Usage Guidelines**

You might want to explicitly send resets for inbound traffic if you need to reset identity request (IDENT) connections. When you send a TCP RST (reset flag in the TCP header) to the denied host, the RST stops the incoming IDENT process so that you do not have to wait for IDENT to time out. Waiting for IDENT to time out can cause traffic to slow because outside hosts keep retransmitting the SYN until the IDENT times out, so the **service resetinbound** command might improve performance.

**Examples**

The following example disables outbound resets for all interfaces except for the inside interface:

```
ciscoasa(config)# no
            service resetoutbound
ciscoasa(config)# service resetoutbound interface inside
```

The following example enables inbound resets for all interfaces except for the DMZ interface:

```
ciscoasa(config)# service resetinbound
ciscoasa(config)# no
            service resetinbound interface dmz
```

The following example enables resets for connections that terminate on the outside interface:

```
ciscoasa(config)# service resetoutside
```

**Related Commands**

| Command | Description |
|---|---|
| **show running-config service** | Displays the service configuration. |

# service (object service)

To define the protocol and optional attributes for a service object, use the **service** command in object service configuration mode. Use the **no** form of this command to remove the definition.

**service** { *protocol* | { **tcp | udp | sctp** } [ **source** *operator number* ] [ **destination** *operator number* ] | { **icmp | icmp6** } [ *icmp_type* [ *icmp_code* ] ] }
**no service** { *protocol* | { **tcp | udp | sctp** } [ **source** *operator number* ] [ **destination** *operator number* ] | { **icmp | icmp6** } [ *icmp_type* [ *icmp_code* ] ] }

## Syntax Description

| | |
|---|---|
| **destination** *operator number* | (Optional; **tcp** , **udp** , **sctp** only.) Specifies the destination port name or number, between 0 and 65535. For a list of supported names, see the CLI help. Operators include:<br>• **eq** —Equals the port number.<br>• **gt** —Greater than the port number.<br>• **lt** —Less than the port number.<br>• **neq** —Not equal to the port number.<br>• **range** —A range of ports. Specify two numbers separated by a space, such as **range 1024 4500** . |
| { **icmp** \| **icmp6** } [ *icmp_type* [ *icmp_code* ]] | Specifies that the service type is for ICMP or ICMP version 6 connections. You can optionally specify the ICMP type by name or number, between 0 and 255. (For available optional ICMP type names, see the CLI help.) If you specify a type, you can optionally include an ICMP code, between 1 and 255. |
| *protocol* | Identifies the protocol name or number, between 0 and 255. For a list of supported names, see the CLI help. |
| **sctp** | Specifies that the service type is for Stream Control Transmission Protocol (SCTP) connections. |
| **source** *operator number* | (Optional; **tcp** , **udp** , **sctp** only.) Specifies the source port name or number, between 0 and 65535. For a list of supported names, see the CLI help. The operators are the same as those for **destination** . |
| **tcp** | Specifies that the service type is for TCP connections. |
| **udp** | Specifies that the service type is for UDP connections. |

## 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 |
| Object service configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 8.3(1) | This command was added. |
| 9.0(1) | Support for ICMP code was added. |
| 9.5(2) | Support for SCTP was added. |

## Usage Guidelines

You can use service objects by name in other parts of your configuration, for example ACLs (the **access-list** command) and NAT (the **nat** command).

If you configure an existing service object with a different protocol and port, the new configuration replaces the existing protocol and port with the new ones.

## Examples

The following example shows how to create a service object for SSH traffic:

```
ciscoasa(config)# object service SSH
ciscoasa(config-service-object)# service tcp destination eq ssh
```

The following example shows how to create a service object for EIGRP traffic:

```
ciscoasa(config)# object service EIGRP
ciscoasa(config-service-object)# service eigrp
```

The following example shows how to create a service object for traffic coming from port 0 through 1024 to HTTPS:

```
ciscoasa(config)# object service HTTPS
ciscoasa(config-service-object)# service tcp source range 0 1024 destination eq https
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure object** | Clears all objects created. |
| **object-group service** | Configures a service object. |
| **show running-config object service** | Shows the current service object configuration. |

# service call-home

To enable the Call Home service, use the **service call-home** command in global configuration mode. To disable the Call Home service, use the **no** form of this command.

**service call-home**
**no service call-home**

## Syntax Description

This command has no arguments or keywords.

## Command Default

By default, the service Call Home 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 |
| Global configuration | • Yes | • Yes | • Yes | — | • Yes |

## Command History

| Release | Modification |
|---|---|
| 8.2(2) | This command was added. |

## Examples

The following example shows how to enable the Call Home service:

```
ciscoasa(config)# service call-home
```

The following example shows how to disable the Call Home service:

```
hostname(config)# no service call-home
```

## Related Commands

| Command | Description |
|---|---|
| **call-home (global configuration)** | Enters Call Home configuration mode. |
| **call-home test** | Manually sends a Call Home test message. |
| **show call-home** | Displays Call Home configuration information. |

# service-module

To adjust how quickly the system will determine that a service module is no longer responding, use the **service-module** command in global configuration mode. To return to the default settings, use the **no** form of this command.

**service-module** { *module_id* | **all** } { **keepalive-counter | keepalive-timeout** } *value*
**no service-module** { *module_id* | **all** } { **keepalive-counter | keepalive-timeout** } *value*

## Syntax Description

| | |
|---|---|
| { *module_id* \| **all** } | Specifies the module whose keepalive values you are adjusting. Specify **all** to adjust them for all modules. Use ? to determine the module IDs that are valid for your system. These are typically:<br>• **1** for the module in the first slot.<br>• **sfr** for the ASA FirePOWER module. |
| **keepalive-counter** *value* | The maximum number of keepalives that can be sent without a response before the module is considered down, from 1-12. |
| **keepalive-timeout** *value* | The length of time between sending keepalive messages, from 4-16 seconds. |

## Command Default

Default count is 6, default timeout is 4.

## 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 |
|---|---|
| 9.12(3) | This command was added. |

## Usage Guidelines

The system periodically checks the service module health status by sending control plane keepalive messages. If there are communication delays caused by high CPU, the system might not get a response quickly enough, and conclude that it had not received a response from the module. The system will then declare the module to be down, when it is in fact functioning normally, and close the communication channel. When configured for high availability, the system will then fail over to the backup unit due to service card failure. If this happens frequently in your setup, extend the keepalive time or count to give the system more time to declare the module failed.

## Examples

The following example shows how to change the keepalive count and timeout:

```
ciscoasa(config)# service-module all keepalive-count 10
ciscoasa(config)# service-module all keepalive-timeout 8
```

# service-object

To add a service or service object to a service object group that is not pre-defined as TCP, UDP, or TCP-UDP, use the service-object command in object-group service configuration mode. To remove a service, use the **no** form of this command.

**service-object** { *protocol* | { **tcp | udp |tcp-udp | sctp** } [ **source** *operator number* ] [ **destination** *operator number* ] | { **icmp | icmp6** } [ *icmp_type* [ *icmp_code* ] ] | **object** *name* }
**no service-object** { *protocol* | { **tcp | udp |tcp-udp | sctp** } [ **source** *operator number* ] [ **destination** *operator number* ] | { **icmp | icmp6** } [ *icmp_type* [ *icmp_code* ] ] | **object** *name* }

## Syntax Description

| | |
|---|---|
| **destination** *operator number* | (Optional; **tcp** , **udp** , **tcp-udp** , **sctp** only.) Specifies the destination port name or number, between 0 and 65535. For a list of supported names, see the CLI help. Operators include:<br>• **eq** —Equals the port number.<br>• **gt** —Greater than the port number.<br>• **lt** —Less than the port number.<br>• **neq** —Not equal to the port number.<br>• **range** —A range of ports. Specify two numbers separated by a space, such as **range 1024 4500** . |
| { **icmp** \| **icmp6** } [ *icmp_type* [ *icmp_code* ]] | Specifies that the service type is for ICMP or ICMP version 6 connections. You can optionally specify the ICMP type by name or number, between 0 and 255. (For available optional ICMP type names, see the CLI help.) If you specify a type, you can optionally include an ICMP code, between 1 and 255. |
| **object** *name* | Adds the named object or group to the object. |
| *protocol* | Identifies the protocol name or number, between 0 and 255. For a list of supported names, see the CLI help. |
| **sctp** | Specifies that the service type is for Stream Control Transmission Protocol (SCTP) connections. |
| **source** *operator number* | (Optional; **tcp** , **udp** , **tcp-udp** , **sctp** only.) Specifies the source port name or number, between 0 and 65535. For a list of supported names, see the CLI help. The operators are the same as those for **destination** . |
| tcp | Specifies that the service type is for TCP connections. |
| **tcp-udp** | Specifies that the service type is for TCP or UDP connections. |
| udp | Specifies that the service type is for UDP connections. |

## 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 |
| Object-group service configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 8.0(1) | This command was added. |
| 8.3(1) | The **object** keyword was added to support service objects (the **object service** command). |
| 9.0(1) | Support for ICMP code was added. |
| 9.5(2) | Support for SCTP was added. |

## Usage Guidelines

When you create a service object group with the **object-group service** command, and you do not pre-define the protocol type for the whole group, then you can add multiple services and service objects to the group of various protocols, including ports, using the **service-object** command. When you create a service object group for a specific protocol type using the **object-group service** [ **tcp** | **udp** | **tcp-udp** ] command, then you can only identify the destination ports for the object group using the **port-object** command.

## Examples

The following example shows how to add both TCP and UDP services to a service object group:

```
ciscoasa(config)# object-group service CommonApps
ciscoasa(config-service-object-group)# service-object tcp destination eq ftp
ciscoasa(config-service-object-group)# service-object tcp-udp destination eq www
ciscoasa(config-service-object-group)# service-object tcp destination eq h323
ciscoasa(config-service-object-group)# service-object tcp destination eq https
ciscoasa(config-service-object-group)# service-object udp destination eq ntp
```

The following example shows how to add multiple service objects to a service object group:

```
hostname(config)# service object SSH
hostname(config-service-object)# service tcp destination eq ssh
hostname(config)# service object EIGRP
hostname(config-service-object)# service eigrp
hostname(config)# service object HTTPS
hostname(config-service-object)# service tcp source range 0 1024 destination eq https
ciscoasa(config)# object-group service Group1
ciscoasa(config-service-object-group)# service-object object SSH
ciscoasa(config-service-object-group)# service-object object EIGRP
ciscoasa(config-service-object-group)# service-object object HTTPS
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure object-group** | Removes all the **object-group** commands from the configuration. |

| Command | Description |
|---|---|
| **network-object** | Adds a network object to a network object group. |
| **object service** | Adds a service object. |
| **object-group** | Defines object groups to optimize your configuration. |
| **port-object** | Adds a port object to a service object group. |
| **show running-config object-group** | Displays the current object groups. |

# service password-recovery

To enable password recovery, use the **service password-recovery** command in global configuration mode. To disable password recovery, use the **no** form of this command. Password recovery is enabled by default, but you might want to disable it to ensure that unauthorized users cannot use the password recovery mechanism to compromise the ASA.

**service password-recovery**
**no service password-recovery**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Password recovery 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 |
|---|---|
| 7.0(1) | This command was added. |

## Usage Guidelines

On the ASA 5500 series adaptive security appliance, if you forget the passwords, you can boot the ASA into ROMMON by pressing the **Escape** key on the terminal keyboard when prompted during startup. Then set the ASA to ignore the startup configuration by changing the configuration register (see the **config-register** command). For example if your configuration register is the default 0x1, then change the value to 0x41 by entering the **confreg 0x41** command. After reloading the ASA, it loads a default configuration, and you can enter privileged EXEC mode using the default passwords. Then load the startup configuration by copying it to the running configuration and reset the passwords. Finally, set the ASA to boot as before by setting the configuration register to the original setting. For example, enter the **config-register 0x1** command in global configuration mode.

On the PIX 500 series security appliance, boot the ASA into monitor mode by pressing the **Escape** key on the terminal keyboard when prompted during startup. Then download the PIX password tool to the ASA, which erases all passwords and **aaa authentication** commands.

On the ASA 5500 series adaptive security appliance, the **no service password-recovery** command prevents a user from entering ROMMON with the configuration intact. When a user enters ROMMON, the ASA prompts the user to erase all Flash file systems. The user cannot enter ROMMON without first performing this erasure. If a user chooses not to erase the Flash file system, the ASA reloads. Because password recovery depends on using ROMMON and maintaining the existing configuration, this erasure prevents you from recovering a password. However, disabling password recovery prevents unauthorized users from viewing the configuration or inserting different passwords. In this case, to recover the system to an operating state, load

a new image and a backup configuration file, if available. The **service password-recovery** command appears in the configuration file for informational purposes only; when you enter the command at the CLI prompt, the setting is saved in NVRAM. The only way to change the setting is to enter the command at the CLI prompt. Loading a new configuration with a different version of the command does not change the setting. If you disable password recovery when the ASA is configured to ignore the startup configuration at startup (in preparation for password recovery), then the ASA changes the setting to boot the startup configuration as usual. If you use failover, and the standby unit is configured to ignore the startup configuration, then the same change is made to the configuration register when the **no service password recovery** command replicates to the standby unit.

On the PIX 500 series security appliance, the **no service password-recovery** command forces the PIX password tool to prompt the user to erase all Flash file systems. The user cannot use the PIX password tool without first performing this erasure. If a user chooses not to erase the Flash file system, the ASA reloads. Because password recovery depends on maintaining the existing configuration, this erasure prevents you from recovering a password. However, disabling password recovery prevents unauthorized users from viewing the configuration or inserting different passwords. In this case, to recover the system to an operating state, load a new image and a backup configuration file, if available.

## Examples

The following example disables password recovery for the ASA 5500 series:

```
ciscoasa(config)# no service password-recovery
WARNING: Executing "no service password-recovery" has disabled the password recovery mechanism
 and disabled access to ROMMON. The only means of recovering from lost or forgotten passwords
 will be for ROMMON to erase all file systems including configuration files and images. You
 should make a backup of your configuration and have a mechanism to restore images from the
 ROMMON command line.
```

The following example for the ASA 5500 series shows when to enter ROMMON at startup and how to complete a password recovery operation.

```
Use BREAK or ESC to interrupt boot.
Use SPACE to begin boot immediately.
Boot interrupted.

Use ? for help.
rommon #0> confreg

Current Configuration Register: 0x00000001
Configuration Summary:
  boot default image from Flash

Do you wish to change this configuration? y/n [n]: n

rommon #1> confreg 0x41

Update Config Register (0x41) in NVRAM...

rommon #2> boot
Launching BootLoader...
Boot configuration file contains 1 entry.

Loading disk0:/ASA_7.0.bin... Booting...
###################
...
Ignoring startup configuration as instructed by configuration register.
Type help or '?' for a list of available commands.
ciscoasa> enable
Password:
```

```
ciscoasa# configure terminal
ciscoasa(config)# copy startup-config running-config

Destination filename [running-config]?
Cryptochecksum(unchanged): 7708b94c e0e3f0d5 c94dde05 594fbee9

892 bytes copied in 6.300 secs (148 bytes/sec)
ciscoasa(config)# enable password
 NewPassword
ciscoasa(config)# config-register 0x1
```

## Related Commands

| Command | Description |
|---|---|
| **config-register** | Sets the ASA to ignore the startup configuration when it reloads. |
| **enable password** | Sets the enable password. |
| **password** | Sets the login password. |

# service-policy (class)

To apply a hierarchical policy map under another policy map, use the **service-policy** command in class configuration mode. To disable the service policy, use the **no** form of this command. Hierarchical policies are supported only for QoS traffic shaping when you want to perform priority queuing on a subset of shaped traffic.

**service-policy** *policymap_name*
**no service-policy** *policymap_name*

## Syntax Description

| | |
|---|---|
| *policymap_name* | Specifies the policy map name that you configured in the **policy-map** command. You can only specify a Layer 3/4 policy map that includes the **priority** command. |

## 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 |
|---|---|
| 7.2(4)/8.0(4) | This command was added. |

## Usage Guidelines

Hierarchical priority queuing is used on interfaces on which you enable a traffic shaping queue. A subset of the shaped traffic can be prioritized. The standard priority queue is not used (the **priority-queue** command).

For hierarchical priority-queuing, perform the following tasks using Modular Policy Framework:

**1.class-map**—Identify the traffic on which you want to perform priority queuing.

**2.policy-map** (for priority queuing)—Identify the actions associated with each class map.

**a.class**—Identify the class map on which you want to perform actions.

**b.priority**—Enable priority queuing for the class map. You can only include the priority command in this policy map if you want to use is hierarchically.

**3.policy-map** (for traffic shaping)—Identify the actions associated with the **class-default** class map.

**a.class class-default**—Identify the **class-default** class map on which you want to perform actions.

**b.shape**—Apply traffic shaping to the class map.

**c.service-policy**—Call the priority queuing policy map in which you configured the **priority** command so you can apply priority queuing to a subset of shaped traffic.

**4.service-policy**—Assigns the policy map to an interface or globally.

## Examples

The following example enables traffic shaping for all traffic on the outside interface, and prioritizes traffic within VPN tunnel-grp1 with the DSCP bit set to ef:

```
ciscoasa
(config)#
class-map TG1-voice
ciscoasa
(config-cmap)#
match tunnel-group tunnel-grp1
ciscoasa
(config-cmap)#
match dscp ef
ciscoasa(config)# policy-map priority-sub-policy
ciscoasa(config-pmap)# class
 TG1-voice
ciscoasa(config-pmap-c)# priority
ciscoasa(config-pmap-c)# policy-map shape_policy
ciscoasa(config-pmap)# class
 class-default
ciscoasa(config-pmap-c)# shape
ciscoasa(config-pmap-c)# service-policy priority-sub-policy
ciscoasa
(config-pmap-c)#
service-policy shape_policy
 interface outside
```

## Related Commands

| Command | Description |
|---|---|
| **class (policy-map)** | Identifies a class map for a policy map. |
| **clear configure service-policy** | Clears service policy configurations. |
| **clear service-policy** | Clears service policy statistics. |
| policy-map | Identifies actions to perform on class maps. |
| **priority** | Enables priority queuing. |
| **service-policy (global)** | Applies a policy map to an interface. |
| **shape** | Enables traffic shaping. |
| **show running-config service-policy** | Displays the service policies configured in the running configuration. |
| **show service-policy** | Displays the service policy statistics. |

# service-policy (global)

To activate a policy map globally on all interfaces or on a targeted interface, use the **service-policy** command in global configuration mode. To disable the service policy, use the **no** form of this command. Use the **service-policy** command to enable a set of policies on an interface.

**service-policy** *policymap_name* [ **global | interface** *intf* ] [ **fail-close** ]
**no service-policy** *policymap_name* [ **global | interface** *intf* ] [ **fail-close** ]

## Syntax Description

| | |
|---|---|
| **fail-close** | Generates a syslog (767001) for IPv6 traffic that is dropped by application inspections that do not support IPv6 traffic. By default, syslogs are not generated. |
| **global** | Applies the policy map to all interfaces. |
| **interface** *intf* | Applies the policy map to a specific interface. |
| *policymap_name* | Specifies the policy map name that you configured in the **policy-map** command. You can only specify a Layer 3/4 policy map, and not an inspection policy map ( **policy-map type inspect** ). |

## 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. |
| 9.0(1) | The **fail-close** keyword was added. |

## Usage Guidelines

To enable the service policy, use the Modular Policy Framework:

1. **class-map** —Identify the traffic on which you want to perform priority queuing.
2. **policy-map** —Identify the actions associated with each class map.

**a. class** —Identify the class map on which you want to perform actions.

**b.** *commands for supported features* —For a given class map, you can configure many actions for various features, including QoS, application inspection, CSC or AIP SSM, TCP and UDP connections limits and timeout, and TCP normalization. See the CLI configuration guide for more details about the commands available for each feature.

3. **service-policy** —Assigns the policy map to an interface or globally.

Interface service policies take precedence over the global service policy for a given feature. For example, if you have a global policy with inspections, and an interface policy with TCP normalization, then both inspections and TCP normalization are applied to the interface. However, if you have a global policy with inspections, and an interface policy with inspections, then only the interface policy inspections are applied to that interface.

By default, the configuration includes a global policy that matches all default application inspection traffic and applies inspection to the traffic globally. You can only apply one global policy, so if you want to alter the global policy, you need to either edit the default policy or disable it and apply a new one.

The default service policy includes the following command:

```
service-policy global_policy global
```

## Examples

The following example shows how to enable the inbound_policy policy map on the outside interface:

```
ciscoasa(config)# service-policy inbound_policy interface outside
```

The following commands disable the default global policy, and enables a new one called new_global_policy on all other ASA interfaces:

```
ciscoasa(config)# no service-policy global_policy global
ciscoasa(config)# service-policy new_global_policy global
```

## Related Commands

| Command | Description |
|---|---|
| **clear configure service-policy** | Clears service policy configurations. |
| **clear service-policy** | Clears service policy statistics. |
| **service-policy (class)** | Applies a hierarchical policy under another policy map. |
| **show running-config service-policy** | Displays the service policies configured in the running configuration. |
| **show service-policy** | Displays the service policy statistics. |

# service sw-reset-button

To enable the reset button on the ASA 5506-X, 5508-X, and 5516-X, use the **service sw-reset-button** command in global configuration mode. To disable the reset button, use the **no** form of this command.

**service sw-reset-button**
**no service sw-reset-button**

## Syntax Description

This command has no arguments or keywords.

## Command Default

By default, **service sw-reset-button** 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 |
|---|---|
| 9.3(2) | Command added. |

## Usage Guidelines

The reset button is a small recessed button on the rear panel that if pressed for longer than three seconds resets the ASA to its default "as-shipped" state following the next reboot. Configuration variables are reset to factory default. However, the flash is not erased, and no files are removed.

## Examples

The following example enables the software reset button:

```
ciscoasa(config)# service sw-reset-button
ciscoasa(config)# show sw-reset-button
Software Reset Button is configured.
```

The following example disables the software reset button:

```
ciscoasa(config)# no service sw-reset-button
ciscoasa(config)# show sw-reset-button
Software Reset Button is not configured.
```

## Related Commands

| Command | Description |
|---|---|
| **show running-config service** | Displays the service configuration. |

# service telemetry

When the telemetry data service is enabled, information about the device information, CPU/memory/disk/bandwidth usage, license usage, configured feature list, cluster/failover information, and the alike on the customer ASA devices are sent to Cisco Security Service Exchange (SSE) through Secure Firewall eXtensible Operating System (FXOS). Use the **service telemetry** command in global configuration mode to enable the service. To disable the telemetry service, use the **no** form of this command.

**service telemetry**
**no service telemetry**

## Syntax Description

This command has no arguments or keywords.

## Command Default

By default, the service telemetry 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 |
|---|---|
| 9.13(1) | This command was introduced. |

## Usage Guidelines

The ASA telemetry service is supported in the SSPXRU (FP9300 and FP4100) platforms running the ASA application. This command is used to control per blade telemetry support. To control per chassis telemetry support, you need to enable it in the FXOS/chassis manager.

## Examples

The following example shows how to enable the telemetry service:

```
ciscoasa(config)# service telemetry
```

The following example shows how to disable the telemetry service:

```
hostname(config)# no service telemetry
```

## Related Commands

| Command | Description |
|---|---|
| **show telemetry** | Displays the past 100 events related to telemetry configuration and activities. Also, displays the last sent telemetry data and samples in JSON format. |

# session

To establish a Telnet session from the ASA to a module, such as an IPS SSP or a CSC SSM, to access the module CLI, use the **session** command in privileged EXEC mode.

**session***id*

## Syntax Description

| | |
|---|---|
| *id* | Specifies the module ID:<br>• Physical module—**1** (for slot number 1)<br>• Software module, ASA FirePOWER—**sfr**<br>• Software module, IPS—**ips**<br>• Software module, ASA CX—**cxsc** |

## 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. |
| 8.6(1) | The **ips** module ID for the IPS SSP software module was added. |
| 9.1(1) | Support for the ASA CX module was added (the **cxsc** keyword). |
| 9.2(1) | Support for the ASA FirePOWER module was added (the **sfr** keyword). |

## Usage Guidelines

This command is only available when the module is in the Up state. See the **show module** command for state information.

To end a session, enter **exit** or **Ctrl-Shift-6,** then the **x** key.

Note that the **session 1** command does not work with the following hardware modules:

- ASA CX
- ASA FirePOWER

## Examples

The following example sessions to a module in slot 1:

```
ciscoasa# session 1
Opening command session with slot 1.
Connected to slot 1. Escape character sequence is 'CTRL-^X'.
```

## Related Commands

| Command | Description |
|---|---|
| **debug session-command** | Shows debugging messages for sessions. |

# session console

To establish a virtual console session from the ASA to a software module, such as an IPS SSP software module, use the **session console** command in privileged EXEC mode. This command might be useful if you cannot establish a Telnet session using the **session** command because the control plane is down.

**session** *id* **console**

## Syntax Description

*id* Specifies the module ID:

- ASA FirePOWER module—**sfr**
- IPS module—**ips**
- ASA CX module—**cxsc**
- ASA 5506W-X wireless access point—**wlan**

## Command Default

No default behavior or values.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 8.6(1) | This command was added. |
| 9.1(1) | Support for the ASA CX module was added (the **cxsc** keyword). |
| 9.2(1) | Support for the ASA FirePOWER module was added (the **sfr** keyword). |
| 9.4(1) | Support for the ASA 5506W-X wireless access point (the **wlan** keyword) was added. |

## Usage Guidelines

To end a session, enter **Ctrl-Shift-6,** then the **x** key.

Do not use this command in conjunction with a terminal server where **Ctrl-Shift-6, x** is the escape sequence to return to the terminal server prompt. **Ctrl-Shift-6, x** is also the sequence to escape the module console and return to the ASA prompt. Therefore, if you try to exit the module console in this situation, you instead exit all the way to the terminal server prompt. If you reconnect the terminal server to the ASA, the module console session is still active; you can never exit to the ASA prompt. You must use a direct serial connection to return the console to the ASA prompt.

Use the **session** command instead.

## Examples

The following example creates a console session to the IPS module:

```
ciscoasa# session ips console
Establishing console session with slot 1
Opening console session with module ips.
Connected to module ips. Escape character sequence is 'CTRL-SHIFT-6 then x'.
sensor login: service
Password: test
```

The following example creates a console session to the wireless access point:

```
ciscoasa# session wlan console
opening console session with module wlan
connected to module wlan. Escape character sequence is 'CTRL-^X'
ap>
```

## Related Commands

| Command | Description |
|---|---|
| **session** | Initiates a Telnet session to a module. |
| **show module log console** | Displays console log information. |

# session do

To establish a Telnet session and perform a command from the ASA to a module, use the **session do** command in privileged EXEC mode.

**session** *id* **do** *command*

## Syntax Description

*id* Specifies the module ID:

- Physical module—**1** (for slot number 1)
- Software module, ASA FirePOWER—**sfr**
- Software module, IPS—**ips**
- Software module, ASA CX—**cxsc**

*command* Performs a command on the module. Supported commands include:

- **setup host ip** *ip_address*/*mask*,*gateway_ip* —Sets the management IP address and gateway.
- **get-config**—Gets the module configuration.
- **password-reset**—Resets the module password to the default.

## 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 |
|---|---|
| 7.1(1) | This command was added. |
| 8.6(1) | The **ips** module ID for the IPS SSP software module was added. |
| 8.4(4.1) | Support for the ASA CX module was added. |
| 9.2(1) | Support for the ASA FirePOWER module, including the **sfr** keyword was added. |

## Usage Guidelines

This command is only available when the module is in the Up state. See the **show module** command for state information.

To end a session, enter **exit** or **Ctrl-Shift-6,** then the **X** key.

## Examples

The following example sets the management IP address to 10.1.1.2/24, with a default gateway of 10.1.1.1:

```
ciscoasa# session 1 do setup host ip 10.1.1.2/24,10.1.1.1
```

## Related Commands

| Command | Description |
|---|---|
| **debug session-command** | Shows debugging messages for sessions. |

# session ip

To configure logging IP addresses for the module, such as an IPS SSP or a CSC SSM, use the **session ip** command in privileged EXEC mode.

**session** *id* **ip** { **address** *address mask* | **gateway** *address* }

## Syntax Description

| | |
|---|---|
| *id* | Specifies the module ID:<br>• Physical module—**1** (for slot number 1)<br>• Software module, IPS—**ips** |
| **address** *address* | Sets the syslog server address. |
| **gateway** *address* | Sets the gateway to the syslog server. |
| *mask* | Sets the subnet mask. |

## 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 |
|---|---|
| 7.1(1) | This command was added. |
| 8.4(4.1) | Support for the ASA CX module was added. |
| 8.6(1) | The **ips** module ID for the IPS SSP software module was added. |

## Usage Guidelines

This command is only available when the module is in the Up state. See the **show module** command for state information.

To end a session, enter **exit** or **Ctrl-Shift-6,** then the **X** key.

## Examples

The following example sessions to a module in slot 1:

```
ciscoasa# session 1 ip
address
```

## Related Commands

| Command | Description |
|---|---|
| **debug session-command** | Shows debugging messages for sessions. |

# set adaptive-interface cost

To set the output interface based on the adaptive interface cost on the candidate interfaces, use the **set adaptive-interface cost** command in route map configuration mode

**set adaptive-interface cost** *interface_list*

## Syntax Description

| | |
|---|---|
| *interface_list* | A space-separated list of interface names. The egress interface is selected from these interfaces. |

## Command Default

No default values.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.17(1) | This command was introduced. |

## Usage Guidelines

Set the cost of the interface in the interface configuration using the **policy-route cost** command. The default cost is 0, so you can use adaptive interface cost even without setting an explicit cost value.

If the costs of the interfaces are the same, it is an active-active configuration and packets are load-balanced (round-robin) on the egress interfaces. If the costs are different, the interface with the lowest cost is selected. Interfaces are considered only if they are up.

For example, by setting the same cost on 2 WAN links, you can load balance the traffic across those links to perhaps improve performance. However, if one WAN link has higher bandwidth than the other, you can set the higher bandwidth link's cost to 1, and the lower bandwidth link to 2, so that the lower bandwidth link is used only if the higher bandwidth link is down.

After you configure the route map with this command, you must apply it to the ingress interfaces using the **policy-route route-map** command.

### Example

The following example sets output1 and output2 as the candidate egress interfaces based on their cost.

```
ciscoasa(config)# route-map mymap 10
ciscoasa(config-route-map)# match ip address DIA_traffic
ciscoasa(config-route-map)# set adaptive-interface cost output1 output2
```

# set as-path

To modify an autonomous system path for BGP routes, use the set as-path command in route-map configuration mode. To not modify the autonomous system path, use the no form of this command.

**set as-path** { **tag | prepend** *as-path-string* }
**no set as-path** { **tag | prepend** *as-path-string* }

## Syntax Description

| | |
|---|---|
| *as-path-string* | Number of an autonomous system to prepend to the AS_PATH attribute. The range of values for this argument is any valid autonomous system number from 1 to 65535. Multiple values can be entered; up to 10 AS numbers can be entered.<br>For more details about autonomous system number formats, see the router bgp command. |
| **prepend** | Appends the string following the keyword prepend to the autonomous system path of the route that is matched by the route map. Applies to inbound and outbound BGP route maps. |
| **tag** | Converts the tag of a route into an autonomous system path. Applies only when redistributing routes into BGP. |

## Command Default

An autonomous system path is not modified.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

The only global BGP metric available to influence the best path selection is the autonomous system path length. By varying the length of the autonomous system path, a BGP speaker can influence the best path selection by a peer further away.

By allowing you to convert the tag into an autonomous system path, the set as-path tag variation of this command modifies the autonomous system length. The set as-path prepend variation allows you to "prepend" an arbitrary autonomous system path string to BGP routes. Usually the local autonomous system number is prepended multiple times, increasing the autonomous system path length.

Cisco implementation of 4-byte autonomous system numbers uses asplain—65538 for example—as the default regular expression match and output display format for autonomous system numbers, but you can configure 4-byte autonomous system numbers in both the asplain format and the asdot format as described in RFC 5396. To change the default regular expression match and output display of 4-byte autonomous system numbers to

asdot format, use the bgp asnotation dot command followed by the clear bgp * command to perform a hard reset of all current BGP sessions.

## Examples

The following example converts the tag of a redistributed route into an autonomous system path:

```
ciscoasa(config)# route-map set-as-path-from-tag
ciscoasa(config-route-map)# set as-path tag
ciscoasa(config-route-map)# router bgp 100
ciscoasa(config-router)# address-family ipv4
 ciscoasa(config-router-af)# redistribute ospf 109 route-map set-as-path-from-tag
```

The following example prepends 100 100 100 to all the routes that are advertised to 10.108.1.1:

```
ciscoasa(config)# route-map set-as-path
ciscoasa(config-route-map)# match as-path 1
ciscoasa(config-route-map)# set as-path prepend 100 100 100
ciscoasa(config-route-map)# router bgp 100
ciscoasa(config-router)# address-family ipv4
ciscoasa(config-router-af)# neighbor 10.108.1.1 route-map set-as-path out
```

## Related Commands

| Command | Description |
|---|---|
| **clear bgp** | Resets BGP connections using hard or soft reconfiguration. |
| **bgp asnotation dot** | Changes the default display and regular expression match format of Border Gateway Protocol (BGP) 4-byte autonomous system numbers from asplain format (decimal values) to dot notation. |

# set automatic-tag

To automatically compute the tag value, use the **set automatic-tag** command in route-map configuration mode. To disable this function, use the **no** form of this command.

**set automatic-tag**
**no set automatic-tag**

## Syntax Description

This command has no arguments or keywords.

## 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 |
| Route-map configuration | • Yes | — | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

You must have a match clause (even if it points permit everything) if you want to set tags.

Use the **route-map** global configuration command and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *match criteria* --the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the *set actions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **no route-map** command deletes the route map.

The **set** route-map configuration commands specify the redistribution *set actions* to be performed when all the match criteria of a route map are met. When all match criteria are met, all set actions are performed.

## Examples

The following example configures the ASA software to automatically compute the tag value for the Border Gateway Protocol (BGP) learned routes:

```
ciscoasa(config-route-map)# route-map tag
ciscoasa(config-route-map)# match as-path 10
iscoasa(config-route-map)# set automatic-tag
ciscoasa(config-route-map)# router bgp 100
ciscoasa(config-router)# address-family ipv4
ciscoasa(config-router-af)# table-map tag
```

# set community

To set the BGP communities attribute, use the set community route map configuration command. To delete the entry, use the no form of this command.

**set community** { *community-number* [ **additive** ] | [ *well-known-community* ] [ **additive** ] | **none** }
**no set community**

## Syntax Description

| | |
|---|---|
| additive | (Optional) Adds the community to the already existing community. |
| community-number | Specifies that community number. Valid values are from 1 to 4294967200, no-export, or no-advertise. |
| none | (Optional) Removes the community attribute from the prefixes that pass the route map. |
| well-known-community | (Optional) Well-known communities can be specified by using the following keywords:<br>• **internet**<br>• **local-as**<br>• **no-advertise**<br>• **no-export** |

## Command Default

No BGP communities attributes exist.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

You must have a match clause (even if it points to a “permit everything” list) if you want to set tags.

Use the route-map global configuration command, and the match and set route map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which redistribution is allowed for the current route-map command. The set

commands specify the set actions—the particular redistribution actions to perform if the criteria enforced by the match commands are met. The no route-map command deletes the route map.

The set route map configuration commands specify the redistribution set actions to be performed when all of the match criteria of a route map are met. When all match criteria are met, all set actions are performed.

## Examples

In the following example, routes that pass the autonomous system path access list 1 have the community set to 109. Routes that pass the autonomous system path access list 2 have the community set to no-export (these routes will not be advertised to any external BGP [eBGP] peers).

```
ciscoasa(config-route-map)# set community 10
ciscoasa(config-route-map)# match as-path 1
ciscoasa(config-route-map)# set community 109
ciscoasa(config-route-map)# set community 20
ciscoasa(config-route-map)# match as-path 2
ciscoasa(config-route-map)# set community no-export
```

## Related Commands

| Command | Description |
|---|---|
| **match as-path** | Match a BGP autonomous system path that is specified by an access list. |

# set connection

To specify connection limits within a policy map for a traffic class, use the **set connection** command in class configuration mode. To remove these specifications, thereby allowing unlimited connections, use the **no** form of this command.

**set connection** { [ **conn-max** *n* ] [ **embryonic-conn-max** *n* ] [ **per-client-embryonic-max** *n* ] [ **per-client-max** *n* ] [ **syn-cookie-mss** *n* ] [ **random-sequence-number** { **enable | disable** } ] }
**no set connection** { [ **conn-max** *n* ] [ **embryonic-conn-max** *n* ] [ **per-client-embryonic-max** *n* ] [ **per-client-max** *n* ] [ **syn-cookie-mss** *n* ] [ **random-sequence-number** { **enable | disable** } ] }

## Syntax Description

| | |
|---|---|
| **conn-max** *n* | (TCP, UDP, SCTP.) Sets the maximum number of simultaneous connections that are allowed, between 0 and 2000000. The default is 0, which allows unlimited connections. For example, if two servers are configured to allow simultaneous connections, the connection limit is applied to each configured server separately. For TCP connections, this applies to established connections only.<br><br>When configured under a class, this argument restricts the maximum number of simultaneous connections that are allowed for the entire class. In this case, one attack host can consume all the connections and leave none of the rest of the hosts matched in the access list under the class. |
| **embryonic-conn-max** *n* | Sets the maximum number of simultaneous embryonic TCP connections allowed, between 0 and 2000000. The default is 0, which allows unlimited connections. |
| **per-client-embryonic-max** *n* | Sets the maximum number of simultaneous embryonic TCP connections allowed per client, between 0 and 2000000. A client is defined as the host that sends the initial packet of a connection (that builds the new connection) through the ASA.<br><br>If an **access-list** is used with a **class-map** to match traffic for this feature, the embryonic limit is applied per-host, and not the cumulative embryonic connections of all clients that match the access list. The default is 0, which allows unlimited connections. This keyword is not available for management class maps. |
| **per-client-max** *n* | (TCP, UDP, SCTP.) Sets the maximum number of simultaneous connections allowed per client, between 0 and 2000000. A client is defined as the host that sends the initial packet of a connection (that builds the new connection) through the ASA. For TCP connections, this includes established, half-open, and half-closed connections.<br><br>If an **access-list** is used with a **class-map** to match traffic for this feature, the connection limit is applied per-host, and not the cumulative connections of all clients that match the access list. The default is 0, which allows unlimited connections.<br><br>This keyword is not available for management class maps. When configured under a class, this keyword restricts the maximum number of simultaneous connections that are allowed for each host that is matched through an access list under the class. |

| | |
|---|---|
| **random-sequence-number** { **enable** \| **disable** } | Enables or disables TCP sequence number randomization. This keyword is not available for management class maps. See the "Usage Guidelines" section for more information. |
| **syn-cookie-mss** *n* | Sets the server maximum segment size (MSS) for SYN-cookie generation for embryonic connections upon reaching the embryonic connections limit, from 48 to 65535 . The default is 1380. This setting is meaningful only if you configure **set connection embryonic-conn-max** or **per-client-embryonic-max**. |

## Command Default

For the **conn-max** , **embryonic-conn-max** , **per-client-embryonic-max** , and **per-client-max** parameters, the default value of *n* is 0, which allows unlimited connections.

Sequence number randomization 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 |
| Class configuration | • Yes | • Yes | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 7.1(1) | The **per-client-embryonic-max** and **per-client-max** keywords were added. |
| 8.0(2) | This command is now available for a Layer 3/4 management class map, for to-the-ASA management traffic. Only the **conn-max** and **embryonic-conn-max** keywords are available. |
| 9.0(1) | The maximum number of connections was increased from 65535 to 2000000. |
| 9.5(2) | The **conn-max** and **per-client-max** keywords now apply to SCTP as well as TCP and UDP. |
| 9.16(1) | The **syn-cookie-mss** keyword was added. |

## Usage Guidelines

Configure this command using Modular Policy Framework. First define the traffic to which you want to apply the timeout using the **class-map** command (for through traffic) or **class-map type management** command (for management traffic). Then enter the **policy-map** command to define the policy, and enter the **class** command to reference the class map. In class configuration mode, you can enter the **set connection** command. Finally, apply the policy map to an interface using the **service-policy** command. For more information about how Modular Policy Framework works, see the CLI configuration guide.

**Note** Depending on the number of CPU cores on your ASA model, the maximum concurrent and embryonic connections may exceed the configured numbers due to the way each core manages connections. In the worst case scenario, the ASA allows up to *n* -1 extra connections and embryonic connections, where *n* is the number of cores. For example, if your model has 4 cores, if you configure 6 concurrent connections and 4 embryonic connections, you could have an additional 3 of each type. To determine the number of cores for your model, enter the **show cpu core** command.

**TCP Intercept Overview**

Limiting the number of embryonic connections protects you from a DoS attack. The ASA uses the per-client limits and the embryonic connection limit to trigger TCP Intercept, which protects inside systems from a DoS attack perpetrated by flooding an interface with TCP SYN packets. An embryonic connection is a connection request that has not finished the necessary handshake between source and destination. TCP Intercept uses the SYN cookies algorithm to prevent TCP SYN-flooding attacks. A SYN-flooding attack consists of a series of SYN packets usually originating from spoofed IP addresses. The constant flood of SYN packets keeps the server SYN queue full, which prevents it from servicing connection requests. When the embryonic connection threshold of a connection is crossed, the ASA acts as a proxy for the server and generates a SYN-ACK response to the client SYN request. When the ASA receives an ACK back from the client, it can then authenticate the client and allow the connection to the server.

**TCP Sequence Randomization**

Each TCP connection has two ISNs: one generated by the client and one generated by the server. The ASA randomizes the ISN of the TCP SYN passing in both the inbound and outbound directions.

Randomizing the ISN of the protected host prevents an attacker from predicting the next ISN for a new connection and potentially hijacking the new session.

TCP initial sequence number randomization can be disabled if required. For example:

- If another in-line firewall is also randomizing the initial sequence numbers, there is no need for both firewalls to be performing this action, even though this action does not affect the traffic.
- If you use eBGP multi-hop through the ASA, and the eBGP peers are using MD5. Randomization breaks the MD5 checksum.
- You use a WAAS device that requires the ASA not to randomize the sequence numbers of connections.

## Examples

The following is an example of the use of the **set connection** command configure the maximum number of simultaneous connections as 256 and to disable TCP sequence number randomization:

```
ciscoasa(config)# policy-map localpolicy1
ciscoasa(config-pmap)# class local_server
ciscoasa(config-pmap-c)# set connection conn-max 256 random-sequence-number disable
ciscoasa(config-pmap-c)#
```

You can enter this command with multiple parameters or you can enter each parameter as a separate command. The ASA combines the commands into one line in the running configuration. For example, if you entered the following two commands in class configuration mode:

```
ciscoasa(config-pmap-c)# set connection conn-max 600
ciscoasa(config-pmap-c)# set connection embryonic-conn-max 50
```

The output of the **show running-config policy-map** command would display the result of the two commands in a single, combined command:

```
set connection conn-max 600 embryonic-conn-max 50
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Specifies a class-map to use for traffic classification. |
| **clear configure policy-map** | Removes all policy-map configuration, except that if a policy-map is in use in a service-policy command, that policy-map is not removed. |
| **policy-map** | Configures a policy; that is, an association of a traffic class and one or more actions. |
| **show running-config policy-map** | Displays all current policy-map configurations. |
| **show service-policy** | Displays service policy configuration. Use the **set connection** keyword to view policies that include the **set connection** command. |

# set connection advanced-options

To configure advanced connection settings, use the **set connection advanced-options** command in class configuration mode. To remove the options, use the **no** form of this command.

**set connection advanced-options** { *tcp_mapname* | **tcp-state-bypass | sctp-state-bypass | flow-offload** }
**no set connection advanced-options** { *tcp_mapname* | **tcp-state-bypass | sctp-state-bypass | flow-offload** }

## Syntax Description

| | |
|---|---|
| **flow-offload** | Identify matching flows as eligible for off-loading from the ASA and switched directly in the NIC. This provides improved performance for large data flows in data centers. Flow off-load is available for the Firepower 9300 series running FXOS 1.1.3+, or the Firepower 4100 series running FXOS 1.1.4+, or the Secure Firewall 3100 series.<br>You must also enable flow off-loading before this option works. Use the **flow-offload enable** command. |
| **sctp-state-bypass** | Implements SCTP State Bypass to turn off SCTP stateful inspection. SCTP traffic is not validated for protocol conformance. |
| *tcp_mapname* | Name of a TCP map created by the **tcp-map** command. Use this option to customize TCP normalization. |
| **tcp-state-bypass** | Bypass TCP state checking if you use asymmetrical routing in your network. See the Usage section below for detail information and guidelines for using TCP State Bypass. |

## Command Default

No default behavior or values. No options are enabled by default, although all TCP Normalizer options (within a TCP map) have default settings.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 8.2(1) | The **tcp-state-bypass** keyword was added. |
| 9.5(2) | The **sctp-state-bypass** keyword was added. |
| 9.5(2) | The **flow-offload** keyword was added. The option also requires FXOS 1.1.3+, and is available for the Firepower 9300 series. |

| Release | Modification |
|---|---|
| 9.6(1) | Flow offload support was added for the Firepower 4100 series running FXOS 1.1.4+. |
| 9.19(1) | Support added for the Secure Firewall 3100. |

## Usage Guidelines

To customize TCP normalization with a TCP map, use the Modular Policy Framework:

1. **tcp-map** —Identify the TCP normalization actions if you intend to modify them.
2. **class-map** —Identify the traffic on which you want to perform TCP normalization actions.
3. **policy-map** —Identify the actions associated with the class map.

a. **class** —Identify the class map on which you want to perform actions.

b. **set connection advanced options** —Apply a TCP map or another option to the class map.

4. **service-policy** —Assigns the policy map to an interface or globally.

**TCP State Bypass: Allowing Outbound and Inbound Flows through Separate Devices**

By default, all traffic that goes through the ASA is inspected using the Adaptive Security Algorithm and is either allowed through or dropped based on the security policy. The ASA maximizes the firewall performance by checking the state of each packet (is this a new connection or an established connection?) and assigning it to either the session management path (a new connection SYN packet), the fast path (an established connection), or the control plane path (advanced inspection).

TCP packets that match existing connections in the fast path can pass through the ASA without rechecking every aspect of the security policy. This feature maximizes performance. However, the method of establishing the session in the fast path using the SYN packet, and the checks that occur in the fast path (such as TCP sequence number), can stand in the way of asymmetrical routing solutions: both the outbound and inbound flow of a connection must pass through the same ASA.

For example, a new connection goes to ASA 1. The SYN packet goes through the session management path, and an entry for the connection is added to the fast path table. If subsequent packets of this connection go through ASA 1, then the packets will match the entry in the fast path, and are passed through. But if subsequent packets go to ASA 2, where there was not a SYN packet that went through the session management path, then there is no entry in the fast path for the connection, and the packets are dropped.

If you have asymmetric routing configured on upstream routers, and traffic alternates between two ASAs, then you can configure TCP state bypass for specific traffic. TCP state bypass alters the way sessions are established in the fast path and disables the fast path checks. This feature treats TCP traffic much as it treats a UDP connection: when a non-SYN packet matching the specified networks enters the ASA, and there is not a fast path entry, then the packet goes through the session management path to establish the connection in the fast path. Once in the fast path, the traffic bypasses the fast path checks.

**Unsupported Features for TCP State Bypass**

The following features are not supported when you use TCP state bypass:

- Application inspection—Application inspection requires both inbound and outbound traffic to go through the same ASA, so application inspection is not supported with TCP state bypass.
- AAA authenticated sessions—When a user authenticates with one ASA, traffic returning via the other ASA will be denied because the user did not authenticate with that ASA.
- TCP Intercept, maximum embryonic connection limit, TCP sequence number randomization—The ASA does not keep track of the state of the connection, so these features are not applied.

- TCP normalization—The TCP normalizer is disabled.
- SSM functionality—You cannot use TCP state bypass and any application running on an SSM, such as IPS or CSC.

### NAT Guidelines for TCP State Bypass

Because the translation session is established separately for each ASA, be sure to configure static NAT on both ASAs for TCP state bypass traffic; if you use dynamic NAT, the address chosen for the session on ASA 1 will differ from the address chosen for the session on ASA 2.

### Connection Timeout Guidelines for TCP State Bypass

Starting with release 9.10(1), if there is no traffic on a given connection for 2 minutes, the connection times out. You can override this default using the **set connection timeout idle** command. Normal TCP connections timeout by default after 60 minutes. In releases prior to 9.10(1), the TCP state bypass connections use the global timeout value of 60 minutes.

## Examples

The following example shows the use of the **set connection advanced-options** command to specify the use of a TCP map named localmap:

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

The following is an example configuration for TCP state bypass:

```
ciscoasa(config)# access-list tcp_bypass extended permit tcp 10.1.1.0 255.255.255.224 any
ciscoasa(config)# class-map tcp_bypass
ciscoasa(config-cmap)# description "TCP traffic that bypasses stateful firewall"
ciscoasa(config-cmap)# match access-list tcp_bypass
ciscoasa(config-cmap)# policy-map tcp_bypass_policy
ciscoasa(config-pmap)# class tcp_bypass
ciscoasa(config-pmap-c)# set connection advanced-options tcp-state-bypass
ciscoasa(config-pmap-c)# service-policy tcp_bypass_policy interface outside
```

The following is an example configuration for SCTP state bypass:

```
ciscoasa(config)# access-list sctp_bypass extended permit sctp
            10.1.1.0 255.255.255.224 any
ciscoasa(config)# class-map sctp_bypass
ciscoasa(config-cmap)# description "SCTP traffic that bypasses stateful inspection"
ciscoasa(config-cmap)# match access-list sctp_bypass
ciscoasa(config-cmap)# policy-map sctp_bypass_policy
ciscoasa(config-pmap)# class sctp_bypass
ciscoasa(config-pmap-c)# set connection advanced-options sctp-state-bypass
ciscoasa(config-pmap-c)# service-policy sctp_bypass_policy interface outside
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Identifies a class map in the policy map. |
| **class-map** | Creates a class map for use in a service policy. |
| **flow-offload** | Enables flow offload. |
| **policy-map** | Configures a policy map that associates a class map and one or more actions. |
| service-policy | Assigns a policy map to an interface. |
| **set connection timeout** | Sets the connection timeouts. |
| **show running-config policy-map** | Display all current policy-map configurations. |
| **tcp-map** | Creates a TCP map. |

# set connection decrement-ttl

To decrement the time to live value within a policy map for a traffic class, use the **set connection decrement-ttl** command in class configuration mode. To not decrement the time to live, use the **no** form of this command.

**set connection decrement-ttl**
**no set connection decrement-ttl**

## Syntax Description

This command has no arguments or keywords.

## Command Default

By default, the ASA does not decrement the time to live.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.2(2) | This command was added. |

## Usage Guidelines

This command, along with the **icmp unreachable** command, is required to allow a traceroute through the ASA that shows the ASA as one of the hops.

If you decrement time to live, packets with a TTL of 1 will be dropped, but a connection will be opened for the session on the assumption that the connection might contain packets with a greater TTL. Note that some packets, such as OSPF hello packets, are sent with TTL = 1, so decrementing time to live can have unexpected consequences.

## Examples

The following example enables time to live decrements and sets the ICMP unreachable rate limit:

```
ciscoasa(config)# policy-map localpolicy1
ciscoasa(config-pmap)# class local_server
ciscoasa(config-pmap-c)# set connection decrement-ttl
ciscoasa(config-pmap-c)# exit
ciscoasa(config)# icmp unreachable rate-limit 50 burst-size 6
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Specifies a class map to use for traffic classification. |
| **icmp unreachable** | Controls the rate at which ICMP unreachables are allowed through the ASA. |

| Command | Description |
|---|---|
| **policy-map** | Configures a policy; that is, an association of a traffic class and one or more actions. |
| **show running-config policy-map** | Displays all current policy map configurations. |
| **show service-policy** | Displays service policy configuration. |

# set connection timeout

To specify connection timeouts within a policy map for a traffic class, use the **set connection timeout** command in class configuration mode. To remove the timeout, use the **no** form of this command.

**set connection timeout** { [ **embryonic** *hh* **:** *mm* **:** *ss* ] [ **idle** *hh* **:** *mm* **:** *ss* [ **reset** ] ] [ **half-closed** *hh* **:** *mm* **:** *ss* ] [ **dcd** [ *retry_interval* [ *max_retries* ] ] ] }
**no set connection timeout** { [ **embryonic** *hh* **:** *mm* **:** *ss* ] [ **idle** *hh* **:** *mm* **:** *ss* ] [ **reset** ] [ **half-closed** *hh* **:** *mm* **:** *ss* ] [ **dcd** [ *retry_interval* [ *max_retries* ] ] ] }

## Syntax Description

| | |
|---|---|
| **dcd** [ *retry_interval* [ *max_retries* ]] | Enables dead connection detection (DCD). DCD detects a dead connection and allows it to expire, without expiring connections that can still handle traffic. You configure DCD when you want idle, but valid connections to persist. After a TCP connection times out, the ASA sends DCD probes to the end hosts to determine the validity of the connection. If one of the end hosts fails to respond after the maximum retries are exhausted, the ASA frees the connection. If both end hosts respond that the connection is valid, the ASA updates the activity timeout to the current time and reschedules the idle timeout accordingly.<br><br>When operating in transparent firewall mode, you must configure static routes for the endpoints. You cannot use DCD in a cluster until version 9.13(1).<br><br>You can configure the following optional values:<br><br>• *retry_interval* —Time duration in *hh* **:** *mm* **:** *ss* format to wait after each unresponsive DCD probe before sending another probe, between 0:0:1 and 24:0:0. The default is 0:0:15.<br><br>For systems that are operating in a cluster or high-availability configuration, we recommend that you do not set the interval to less than one minute (0:1:0). If the connection needs to be moved between systems, the changes required take longer than 30 seconds, and the connection might be deleted before the change is accomplished.<br><br>• *max_retries* —Sets the number of consecutive failed retries for DCD before declaring the connection as dead. The minimum value is 1 and the maximum value is 255. The default is 5. |
| **embryonic** *hh* **:** *mm* **:** *ss* | Sets the timeout period until a TCP embryonic (half-open) connection is closed, between 0:0:5 and 1193:0:0. The default is 0:0:30. You can also set the value to 0, which means the connection never times out. A TCP connection for which a three-way handshake is not complete is an embryonic connection. |
| **half-closed** *hh* **:** *mm* **:** *ss* | Sets the idle timeout period until a half-closed connection is closed, between 0:5:0 (for 9.1(1) and earlier) or 0:0:30 (for 9.1(2) and later) and 1193:0:0. The default is 0:10:0. You can also set the value to 0, which means the connection never times out. Half-closed connections are not affected by DCD. Also, the ASA does not send a reset when taking down half-closed connections. |
| **idle** *hh* **:** *mm* **:** *ss* | Sets the idle timeout period after which an established connection of any protocol closes. The valid range is from 0:0:1 to 1193:0:0. |

| | |
|---|---|
| **reset** | For TCP traffic only, sends a TCP RST packet to both end systems after idle connections are removed. |

## Command Default

Unless you change the default globally using the timeout command, the defaults are:

- The default **embryonic** timeout is 30 seconds.
- The default **half-closed** idle timeout is 10 minutes.
- The default **dcd** *max_retries* value is 5.
- The default **dcd** *retry_interval* value is 15 seconds.
- The default **idle** timeout is 1 hour.
- The default **udp** idle timeout is 2 minutes.
- The default **icmp** idle timeout is 2 seconds.
- The default **esp** and **ha** idle timeout is 30 seconds.
- For all other protocols, the default idle timeout is 2 minutes.
- To never time out, enter 0:0:0.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 7.2(1) | Support for DCD was added. |
| 8.2(2) | The **tcp** keyword was deprecated in favor of the **idle** keyword, which controls the idle timeout for all protocols. |
| 9.1(2) | The minimum **half-closed** value was lowered to 30 seconds (0:0:30). |
| 9.13(1) | The DCD configuration is now supported in a cluster. |

## Usage Guidelines

Configure this command using Modular Policy Framework. First define the traffic to which you want to apply the timeout using the **class-map** command. Then enter the **policy-map** command to define the policy, and enter the **class** command to reference the class map. In class configuration mode, you can enter the **set connection timeout** command. Finally, apply the policy map to an interface using the **service-policy**

command. For more information about how Modular Policy Framework works, see the CLI configuration guide.

The **show service-policy** command to includes counters to show the amount of activity from DCD.

## Examples

The following example sets the connection timeouts for all traffic:

```
ciscoasa(config)# class-map CONNS
ciscoasa(config-cmap)# match any
ciscoasa(config-cmap)# policy-map CONNS
ciscoasa(config-pmap)# class CONNS
ciscoasa(config-pmap-c)# set connection timeout idle 2:0:0 embryonic 0:40:0 half-closed
0:20:0 dcd
ciscoasa(config-pmap-c)# service-policy CONNS interface outside
```

You can enter **set connection** commands with multiple parameters, or you can enter each parameter as a separate command. The ASA combines the commands into one line in the running configuration. For example, if you entered the following two commands in class configuration mode:

```
ciscoasa(config-pmap-c)# set connection timeout idle 2:0:0
ciscoasa(config-pmap-c)# set connection timeout embryonic 0:40:0
```

Then the output of the **show running-config policy-map** command would display the result of the two commands in the following single, combined command:

```
set connection timeout idle 2:0:0 embryonic 0:40:0
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Specifies a class-map to use for traffic classification. |
| **clear configure policy-map** | Remove all policy-map configuration, except that if a policy-map is in use in a service-policy command, that policy-map is not removed. |
| **policy-map** | Configures a policy; that is, an association of a traffic class and one or more actions. |
| **set connection** | Configure connection values. |
| **show running-config policy-map** | Display all current policy-map configurations. |
| **show service-policy** | Displays counters for DCD and other service activity. |

# set default interface

The set interface command when used with default option will imply that the first attempt to route the matching traffic has to be done through normal route-lookup by looking up for an explicit route. Only when normal route-lookup fails, PBR will forward the traffic using the interface specified. Since both ‘default’ triggered lookup and the interface option triggered lookup depend on the presence of an explicit route to destination. Always ‘default’ lookup will succeed. When ‘default’ lookup fails, it means there is no explicit route to destination. So, interface action cannot be applied. When “set default interface” is configured, only ‘Null0’ can be configured as interface. When this option is configured, if normal route lookup does not yield an explicit route (non-default route) to the destination, traffic will be dropped.

**set default interface Null0**
**no set default interface Null0**

## Syntax Description

| | |
|---|---|
| interface | Interface to which packets are forwarded. |

## Command Default

There is no default for this command and Null0 interface has to be specified as set action.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

Use this command to provide certain users a different default route. If the ASA has no explicit route for the destination, then it routes the packet to this interface. The first interface specified with the set default interface command that is up is used. The optionally specified interfaces are tried in turn.

Use the ip policy route-map interface configuration command, the route-map global configuration command, and the match and set route-map configuration commands to define the conditions for policy routing packets. The ip policy route-map command identifies a route map by name. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which policy routing occurs. The set commands specify the set actions—the particular routing actions to perform if the criteria enforced by the match commands are met.

In PBR for IPv6, use the ipv6 policy route-map or ipv6 local policy route-map command with match and set route map configuration commands to define conditions for policy routing packets.

The set clauses can be used in conjunction with one another. They are evaluated in the following order:

**1.** set ip next-hop

2. set interface
3. set ip default next-hop
4. set default interface

## Examples

```
(config)# route-map testmap
(config-route-map)# set default interface Null0
(config)# show run route-map
!
route-map testmap permit 10
  set default interface Null0
!
(config)# show route-map testmap
route-map testmap, permit, sequence 10
   Match clauses:
   Set clauses:
                default interface Null0
```

# set dscp

The set dscp command is used to set the QoS bits in the matching IP packets.

**set ip dscp** { **0-63** | **af11** | **af12** | **af13** | **af21** | **af22** | **af23** | **af31** | **af32** | **af33** | **af41** | **af42** | **af43** | **cs1** | **cs2** | **cs3** | **cs4** | **cs5** | **cs6** | **cs7** | **default** | **ef** }
**no set ip dscp**
**set ip dscp** { **0-63** | **af11** | **af12** | **af13** | **af21** | **af22** | **af23** | **af31** | **af32** | **af33** | **af41** | **af42** | **af43** | **cs1** | **cs2** | **cs3** | **cs4** | **cs5** | **cs6** | **cs7** | **default** | **ef** }
**no set ip dscp**

## Syntax Description

| | |
|---|---|
| 0-63 | numeric range of dscp value. |
| af | assured forwarding class |
| ef | expedited forwarding |
| default | |
| cs | |

## Command Default

The DSCP value in the ToS byte is not set.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

Once the DSCP bit is set, other quality of service (QoS) features can then operate on the bit settings.

DSCP and Precedence Values Are Mutually Exclusive

The set dscp command cannot be used with the set precedence command to mark the same packet. The two values, DSCP and precedence, are mutually exclusive. A packet can have one value or the other, but not both.

Precedence Value and Queuing

The network gives priority (or some type of expedited handling) to marked traffic. Typically, you set the precedence value at the edge of the network (or administrative domain); data then is queued according to the precedence. Weighted fair queueing (WFQ) can speed up handling for high-precedence traffic at congestion

points. Weighted Random Early Detection (WRED) ensures that high-precedence traffic has lower loss rates than other traffic during times of congestion.

Use of the “from-field” Packet-marking Category

If you are using this command as part of the Enhanced Packet Marking feature, it can specify the “from-field” packet-marking category to be used for mapping and setting the DSCP value. The “from-field” packet-marking categories are as follows:

- Class of service (CoS)
- QoS group

If you specify a “from-field” category but do not specify the table keyword and the applicable table-map-name argument, the default action will be to copy the value associated with the “from-field” category as the DSCP value. For instance, if you configure the set dscp cos command, the CoS value will be copied and used as the DSCP value.

**Note** The CoS field is a three-bit field, and the DSCP field is a six-bit field. If you configure the set dscp cos command, only the three bits of the CoS field will be used.

If you configure the set dscp qos-group command, the QoS group value will be copied and used as the DSCP value.

The valid value range for the DSCP is a number from 0 to 63. The valid value range for the QoS group is a number from 0 to 99. Therefore, when configuring the set dscp qos-group command, note the following points:

- If a QoS group value falls within both value ranges (for example, 44), the packet-marking value will be copied and the packets will be marked.
- If QoS group value exceeds the DSCP range (for example, 77), the packet-marking value will not be copied and the packet will not be marked. No action is taken.

Set DSCP Values in IPv6 Environments

When this command is used in IPv6 environments, the default match occurs on both IP and IPv6 packets. However, the actual packets set by this function are only those that meet the match criteria of the class map containing this function.

Set DSCP Values for IPv6 Packets Only

To set DSCP values for IPv6 values only, you must also use the match protocol ipv6 command. Without that command, the precedence match defaults to match both IPv4 and IPv6 packets.

Set DSCP Values for IPv4 Packets Only

To set DSCP values for IPv4 values only, you must use the appropriate match ip command. Without this command, the class map may match both IPv6 and IPv4 packets, depending on the other match criteria, and the DSCP values may act upon both types of packets.

## Examples

```
(config)# route-map testmapv4
(config-route-map)# set ip dscp af22
(config)# show run route-map
!
route-map testmapv4 permit 10
```

```
  set ip dscp af22
!
(config)# show route-map testmapv4
route-map testmapv4, permit, sequence 10
   Match clauses:
   Set clauses:
                ip dscp af22
(config)# route-map testmapv6
(config-route-map)# set ipv6 dscp cs6
(config)# show run route-map
!
route-map testmapv6 permit 10
  set ipv6 dscp cs6
!
(config)# show route-map testmap
route-map testmap, permit, sequence 10
   Match clauses:
   Set clauses:
                ipv6 dscp cs6
```

# set ikev1 transform-set

To specify the IPsec IKEv1 proposal for the IPsec profile, use the set ikev1 transform-set command in the IPsec profile configuration mode. Use the no form of this command to remove the IPsec IKEv1 proposal.

**set ikev1 transform-set** *transform-set name*
**no set ikev1 transform-set** *transform-set name*

## Syntax Description

| | |
|---|---|
| *transform-set name* | Specifies the name of the IPsec IKEv1 proposal. |

## 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 |
| IPsec profile configuration | • Yes | • No | • Yes | • No | — |

## Command History

| Release | Modification |
|---|---|
| 9.7(1) | We introduced this command. |

## Examples

The following example specifies the IKEv1 proposal for the IPsec profile:

```
ciscoasa(config)# crypto ipsec profile VTIipsec
ciscoasa(config-ipsec-profile)# set ikev1 transform-set
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ipsec profile** | Creates a new IPsec profile. |
| **responder-only** | Sets the VTI tunnel interface to responder only mode. |
| **set pfs** | Specifies the PFS group to be used in the IPsec profile configuration. |
| **set security-association lifetime** | Specifies the duration of security association in the IPsec profile configuration. This is specified in kilobytes or seconds, or both. |
| set trustpoint | Specifies a trustpoint that defines the certificate to be used while initiating a VTI tunnel connection. |

# set interface

The set interface command is used to configure the interface through which the matching traffic has to be forwarded. It is allowed to configure multiple interfaces in which case they are evaluated in the specified order until a valid up and running interface to forward the packets is found. When the interface name is specified as 'Null0', all traffic matching the route-map will be dropped.

**set interface** [ *...interface* ]
**no set interface** [ *...interface* ]

## Syntax Description

| | |
|---|---|
| interface | Interface to which packets are forwarded. |

## Command Default

No command 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 |
| Route-map configuration | • Yes | — | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

Use the ip policy route-map interface configuration command, the route-map global configuration command, and the match and set route-map configuration commands to define the conditions for policy-routing packets. The ip policy route-map command identifies a route map by name. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which policy routing occurs. The set commands specify the set actions—the particular routing actions to perform if the criteria enforced by the match commands are met.

In PBR for IPv6, use the ipv6 policy route-map or ipv6 local policy route-map command with match and set route-map configuration commands to define conditions for policy-routing packets.

If the first interface specified with the set interface command is down, the optionally specified interfaces are tried in turn.

The set clauses can be used in conjunction with one another. They are evaluated in the following order:

**1.** set ip next-hop

**2.** set interface

**3.** set ip default next-hop

**4.** set default interface

A useful next hop implies an interface. As soon as a next hop and an interface are found, the packet is routed.

## Examples

```
ciscoasa(config)# route-map testmap
ciscoasa(config-route-map)# set interface outside
ciscoasa(config)# show run route-map
!
route-map testmap permit 10
  set interface outside
!
ciscoasa(config)# show route-map testmap
route-map testmap, permit, sequence 10
   Match clauses:
   Set clauses:
               interface outside
```

# set ip df

The set ip df command is used to set the df (do-not-fragment) bit in the matching IP packets..

**set ip df** [ **0** | **1** ]
**no set ip df**

## Syntax Description

| | |
|---|---|
| 0 | Sets the df bit to 0 (clears the df bit), allows packets fragmentation. |
| 1 | Sets the DF bit to 1 which prohibits packet fragmentation. |

## Command Default

There is no default for this command and either 0 or 1 has to be specified as DF bit, in the set action.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

Using Path MTU Discovery (PMTUD) you can determine an MTU value for IP packets that avoids fragmentation. If ICMP messages are blocked by a router, the path MTU is broken and packets with the DF bit set are discarded. Use the set ip df command to clear the DF bit and allow the packet to be fragmented and sent. Fragmentation can slow the speed of packet forwarding on the network but access lists can be used to limit the number of packets on which the DF bit will be cleared.

**Note** Some IP transmitters (notably some versions of Linux) may set the identification field in the IP header (IPid) to zero when the DF bit is set. If the router should clear the DF bit on such a packet and if that packet should subsequently be fragmented, then the IP receiver will probably be unable to correctly reassemble the original IP packet.

## Examples

```
(config)# route-map testmap
(config-route-map)# set ip df 1
(config)# show run route-map
!
route-map testmap permit 10
  set ip df 1
!
(config)# show route-map testmap
```

```
route-map testmap, permit, sequence 10
   Match clauses:
   Set clauses:
                ip df 1
```

# set ip default next-hop

The set ip next-hop command when used with the default option implies that the first attempt to route the matching traffic has to be done through normal route-lookup by looking for an explicit route. Only when normal route-lookup fails, Policy Based Routing (PBR) will forward the traffic using the specified next-hop ip address.

**set ip default next-hop ip-address** [ *...ip-address* ]
**no set ip default next-hop ip-address** [ *...ip-address* ]
**set default ipv6next-hop ip-address** [ *...ip-address* ]
**no set default ipv6next-hop ip-address** [ *...ip-address* ]

## Syntax Description

| | |
|---|---|
| ip-address | IP address of the next hop to which packets are output. It need not be an adjacent router. |
| ipv6-address | IPv6 address of the next hop to which packets are output. It need not be an adjacent router. |

## Command Default

This command is disabled by default and at least one next-hop ip address has to be specified for the set action.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

Use this command to provide certain users a different default route. If the software has no explicit route for the destination in the packet, then it routes the packet to this next hop. The first next hop specified with the set ip default next-hop command needs to be adjacent to the router. The optional specified IP addresses are tried in turn.

Use the ip policy route-map interface configuration command, the route-map global configuration command, and the match and set route-map configuration commands to define the conditions for policy routing packets. The ip policy route-map command identifies a route map by name. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria--the conditions under which policy routing occurs. The set commands specify the set actions--the particular routing actions to perform if the criteria enforced by the match commands are met.

If the first next hop specified with the set next-hop command is down, the optionally specified IP addresses are tried in turn.

The set clauses can be used in conjunction with one another. They are evaluated in the following order:

**1.** set next-hop

2. set interface
3. set default next-hop
4. set default interface

**Note** The set ip next-hop and set ip default next-hop are similar commands but have a different order of operations. Configuring the set ip next-hop command causes the system to use policy routing first and then use the routing table. Configuring the set ip default next-hop command causes the system to use the routing table first and then policy route the specified next hop.

## Examples

```
(config)# route-map testmapv4
(config-route-map)# set ip default next-hop 1.1.1.1
(config)# show run route-map
!
route-map testmapv4 permit 10
  set ip default next-hop 1.1.1.1
!
(config)# show route-map testmapv4
route-map testmapv4, permit, sequence 10
Match clauses:
Set clauses:
ip default next-hop 1.1.1.1
(config)# route-map testmapv6
(config-route-map)# set ipv6 default next-hop 2001::1
(config)# show run route-map
!
route-map testmapv6 permit 10
  set ipv6 default next-hop 2001::1
!
(config)# show route-map testmapv6
route-map testmapv6, permit, sequence 10
Match clauses:
Set clauses:
ipv6 default next-hop 2001::1
```

# set ip next-hop

To indicate where to output packets that pass a match clause of a route map for policy routing, use the set ip next-hop command in route-map configuration mode. To delete an entry, use the no form of this command.

**set ip next-hop** *ip-address* [ *ip-address* ] [ **peer-address** ]
**no set ip next-hop** *ip-address* [ *ip-address* ] [ **peer-address** ]
**set ipv6 next-hop**

## Syntax Description

| | |
|---|---|
| ip-address | IP address of the next hop to which packets are output. It need not be an adjacent router. |
| peer-address | (Optional) Sets the next hop to be the BGP peering address. |

## 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 |
| Route-map configuration | • Yes | — | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

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

Use the ip policy route-map interface configuration command, the route-map global configuration command, and the match and set route-map configuration commands to define the conditions for policy routing packets. The ip policy route-map command identifies a route map by name. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which policy routing occurs. The set commands specify the set actions—the particular routing actions to perform if the criteria enforced by the match commands are met.

If the first next hop specified with the set next-hop command is down, the optionally specified IP addresses are tried in turn.

When the set next-hop command is used with the peer-address keyword in an inbound route map of a BGP peer, the next hop of the received matching routes will be set to be the neighbor peering address, overriding any third-party next hops. So the same route map can be applied to multiple BGP peers to override third-party next hops.

When the set next-hop command is used with the peer-address keyword in an outbound route map of a BGP peer, the next hop of the advertised matching routes will be set to be the peering address of the local router, thus disabling the next hop calculation. The set next-hop command has finer granularity than the (per-neighbor)

neighbor next-hop-self command, because you can set the next hop for some routes, but not others. The neighbor next-hop-self command sets the next hop for all routes sent to that neighbor.

The set clauses can be used in conjunction with one another. They are evaluated in the following order:

**1.** set next-hop

**2.** set interface

**3.** set default next-hop

**4.** set default interface

**Note** To avoid a common configuration error for reflected routes, do not use the set next-hop command in a route map to be applied to BGP route reflector clients.

## Examples

In the following example, three routers are on the same LAN (with IP addresses 10.1.1.1, 10.1.1.2, and 10.1.1.3). Each is in a different autonomous system. The set ip next-hop peer-address command specifies that traffic from the router (10.1.1.3) in remote autonomous system 300 for the router (10.1.1.1) in remote autonomous system 100 that matches the route map is passed through the router bgp 200, rather than sent directly to the router (10.1.1.1) in autonomous system 100 over their mutual connection to the LAN.

```
ciscoasa(config)# router bgp 200
ciscoasa(config-router)# address-family ipv4
ciscoasa(config-router-af)# neighbor 10.1.1.3 remote-as 300
ciscoasa(config-router-af)# neighbor 10.1.1.3 route-map set-peer-address out
ciscoasa(config-router-af)# neighbor 10.1.1.1 remote-as 100
ciscoasa(config-route-af)# route-map set-peer-address permit 10
ciscoasa(config-route-map)# set ip next-hop peer-address
```

# set ip next-hop recursive

Both set ip next-hop and set ip default next-hop require that the next-hop be found on a directly connected subnet. With set ip next-hop recursive, the next-hop address does not need to be directly connected. Instead a recursive lookup is performed on the next-hop address, and matching traffic is forwarded to the next-hop used by that route entry according to the routing path in use on the router.

Recursive next-hop lookup is not applicable for IPv6 or when default keyword is specified.

**set ip next-hop recursive** [ *ipv4-address* ]
**no set ip next-hop recursive** [ *ipv4-address* ]

## Syntax Description

| | |
|---|---|
| ipv4-address | IP address of the next hop to which packets are output. It need not be an adjacent router. |

## 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 |
| Route-map configuration | • Yes | — | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

Use the ip policy route-map interface configuration command, the route-map global configuration command, and the match and set route-map configuration commands to define the conditions for policy routing packets. The ip policy route-map command identifies a route map by name. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which policy routing occurs. The set commands specify the set actions—the particular routing actions to perform if the criteria enforced by the match commands are met.

If the interface associated with the first next hop specified with the set ip next-hop command is down, the optionally specified IP addresses are tried in turn.

The set clauses can be used in conjunction with one another. They are evaluated in the following order:

**1.** set ip next-hop

**2.** set interface

**3.** set ip default next-hop

**4.** set default interface

**Note** The set ip next-hop and set ip default next-hop are similar commands but have a different order of operations. Configuring the set ip next-hop command causes the system to use policy routing first and then use the routing table. Configuring the set ip default next-hop command causes the system to use the routing table first and then policy route the specified next hop.

## Examples

```
(config)# route-map testmapv4
(config-route-map)# set ip next-hop recursive 1.1.1.1
(config)# show run route-map
!
route-map testmapv4 permit 10
  set ip next-hop recursive 1.1.1.1
!
(config)# show route-map testmapv4
route-map testmapv4, permit, sequence 10
   Match clauses:
   Set clauses:
               ip next-hop recursive 1.1.1.1
```

# set ip next-hop verify-availability

The set ip next-hop verify-availability can be configured with an SLA monitor tracking object to verify the reachability of the next-hop. To verify the availability of multiple next-hops, multiple set ip next-hop verify-availability commands can be configured with different sequence numbers and different tracking objects.

**set ip next-hop verify-availability** [ *sequence number* ] **track** [ *tracked-object-number* ]
**no set ip next-hop verify-availability** [ *sequence number* ] **track** [ *tracked-object-number* ]

## Syntax Description

| | |
|---|---|
| sequence-number | Sequence of next hops. The acceptable range is from 1-65535. |
| track | The tracking method is track. |
| tracked-object-number | Object number that the tracking subsystem is tracking. The acceptable range is from 1 to 500. |

## Command Default

No command 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 |
| Route-map configuration | • Yes | — | • Yes | — | — |

## Command History

| Release | Modification |
|---|---|
| 9.4(1) | This command was added. |

## Usage Guidelines

The set ip next-hop verify-availability command can be used in the following two ways:

- With policy-based routing (PBR) to verify next hop reachability using Cisco Discovery Protocol (CDP).
- With optional arguments to support object tracking using Internet Control Message Protocol (ICMP) ping or an HTTP GET request to verify if a remote device is reachable.

Using CDP Verification

This command is used to verify that the next hop is reachable before the router tries to policy route to it. This command has the following characteristics:

- It causes some performance degradation.
- CDP must be configured on the interface.
- The next hop must be a Cisco device with CDP enabled.

- It is supported in process switching and Cisco Express Forwarding (CEF) policy routing, but is not available in distributed CEF (dCEF) because of the dependency of the CDP neighbor database.

If the router is policy routing packets to the next hop and the next hop is down, the router will try unsuccessfully to use Address Resolution Protocol (ARP) for the next hop (which is down). This behavior will continue indefinitely. To prevent this situation from occurring, use the set ip next-hop verify-availability command to configure the router to verify that the next hop of the route map is a CDP neighbor before routing to that next hop.

This command is optional because some media or encapsulations do not support CDP, or it may not be a Cisco device that is sending traffic to the router.

If this command is set and the next hop is not a CDP neighbor, then the router looks to the subsequent next hop, if there is one. If there is no next hop, the packets are not policy routed.

If this command is not set, the packets are either successfully policy routed or remain forever unrouted.

If you want to selectively verify availability of only some next hops, you can configure different route map entries (under the same route map name) with different criteria (using access list matching or packet size matching), and then use the set ip next-hop verify-availability command selectively.

Using Object Tracking

With optional arguments to support object tracking, this command allows PBR to make decisions based on the following criteria:

- ICMP ping reachability to a remote device.
- Application running on a remote device (for example, the device responds to an HTTP GET request).
- A route exists in the Routing Information Base (RIB) (for example, policy route only if 10.2.2.0/24 is in the RIB).
- Interface state (for example, packets received on E0 should be policy routed out E1 only if E2 is down).

Object tracking functions in the following manner. PBR will inform the tracking process that it is interested in tracking a certain object. The tracking process will in turn notify PBR when the state of the object changes. This notification is done via registries and is event driven.

The tracking subsystem is responsible for tracking the state of an object. The object can be an IP address that is periodically being pinged by the tracking process. The state of the object (up or down) is stored in a track report data structure. The tracking process will create the tracking object report. Then the exec process that is configuring the route map can query the tracking process to determine if a given object exists. If the object exists, the tracking subsystem can start tracking it and read the initial state of the object. If the object changes state, the tracking process will notify all the clients that are tracking this process that the state of the object has changed. So, the route map structure that PBR is using can be updated to reflect the current state of the object in the track report. This interprocess communication is done by means of registries and the shared track report.

**Note** If the CDP and object tracking commands are mixed, the tracked next hops will be tried first.

## Examples

```
ciscoasa(config)# sla monitor 1
ciscoasa(config-sla-monitor)# type echo protocol ipIcmpEcho 1.1.1.1 interface outside
```

```
ciscoasa(config)# sla monitor schedule 1 life forever start-time now
ciscoasa(config)# track 1 rtr 1 reachability
ciscoasa(config)#
ciscoasa(config)# route-map testmapv4
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.1 10 track 1
ciscoasa(config)# show run route-map
!
route-map testmapv4 permit 10
  set ip next-hop verify-availability 1.1.1.1 10 track 1
!
ciscoasa(config)# show route-map testmap
route-map testmapv4, permit, sequence 10
   Match clauses:
   Set clauses:
               ip next-hop verify-availability 1.1.1.1 10 track 1
```

# set local-preference

To specify a preference value for the autonomous system path, use the **set local-preference** command in route-map configuration mode. To delete an entry, use the **no**form of this command.

**set local-preference** *number-value*
**no set local-preference** *number-value*

## Syntax Description

| | |
|---|---|
| number-value | Preference value. An integer from 0 to 4294967295. |

## Command Default

Preference value is 100.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

The preference is sent only to all routers in the local autonomous system.

Use the **route-map** global configuration command, and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *match criteria* --the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the *set actions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **no route-map** command deletes the route map.

The **set** route-map configuration commands specify the redistribution *set actions* to be performed when all the match criteria of a route map are met. When all match criteria are met, all set actions are performed.

You can change the default preference value with the bgp default local-preference command.

## Examples

The following example sets the local preference to 100 for all routes that are included in access list 1:

```
ciscoasa(config-route-map)# route-map map-preference
ciscoasa(config-route-map)# match as-path 1
ciscoasa(config-route-map)# set local-preference 100
```

# set metric

To set the metric value of a route for OSPF and other dynamic routing protocols in a route map, use the set metric command in route-map configuration mode. To return to the default metric value for OSPF and other dynamic routing protocols, use the **no** form of this command.

**set metric** *metric-value* | [ *bandwidth delay reliability loading mtu* ]
**no set metric** *metric-value* | [ *bandwidth delay reliability loading mtu* ]

## Syntax Description

| | |
|---|---|
| *bandwidth* | EIGRP bandwidth of a route, in kbps. Valid values range from 0 to 4294967295. |
| *delay* | EIGRP route delay, in tens of microseconds. Valid values range from 0 to 4294967295. |
| *loading* | Effective EIGRP bandwidth of a route expressed as a number from 0 to 255. The value 255 means 100 percent loading. |
| *metric-value* | Metric value of a route for OSPF and other dynamic routing protocols (except for EIGRP), expressed as a number. Valid values range from 0 to 4294967295. |
| *mtu* | Minimum MTU size of a route for EIGRP, in bytes. Valid values range from 0 to 4294967295. |
| *reliability* | Likelihood of successful packet transmission for EIGRP expressed as a number from 0 to 255. The value 255 means 100 percent reliability; 0 means no reliability. |

## 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 |
| Route-map configuration | • Yes | — | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 8.2(5) | The *bandwidth* , *delay*, *reliability*, *loading*, and *mtu* arguments to support EIGRP in a route map were added. |
| 9.0(1) | Support for multiple context mode was added. |

## Usage Guidelines

The **no** set metric command allows you to return to the default metric value for OSPF and other dynamic routing protocols. In this context, the *metric-value* argument is an integer from 0 to 4294967295.

## Examples

The following example shows how to configure a route map for OSPF routing:

```
ciscoasa(config)# route-map maptag1 permit 8
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# match metric 5
ciscoasa(config-route-map)# show route-map
route-map maptag1 permit 8
set metric 5
match metric 5
```

The following example shows how to set the metric value for EIGRP in a route map:

```
ciscoasa(config)# access-list route-out line 1 standard permit 10.1.1.0 255.255.255.0
ciscoasa(config)# route-map rmap permit 10
ciscoasa(config-route-map)# set metric 10000 60 100 1 1500
ciscoasa(config-route-map)# show route-map rmap
route-map rmap, permit, sequence 10
  Match clauses:
    ip address (access-lists): route-out
  Set clauses:
    metric 10000 60 100 1 1500
ciscoasa(config-route-map)# show running-config route-map
route-map rmap permit 10
 match ip address route-out
 set metric 10000 60 100 1 1500
```

## Related Commands

| Command | Description |
|---|---|
| **match interface** | Distributes any routes that have their next hop out of one of the interfaces specified, |
| **match ip next-hop** | Distributes any routes that have a next-hop router address that is passed by one of the access lists specified. |
| **route-map** | Defines the conditions for redistributing routes from one routing protocol into another. |

# set metric-type

To specify the type of OSPF metric routes, use the set metric-type command in route-map configuration mode. To return to the default setting, use the **no** form of this command.

**set metric-type** { **type-1 | type-2** }
**no set metric-type**

## Syntax Description

| | |
|---|---|
| **type-1** | Specifies the type of OSPF metric routes that are external to a specified autonomous system. |
| **type-2** | Specifies the type of OSPF metric routes that are external to a specified autonomous system. |

## Command Default

The default is type-2.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 9.0(1) | Support for multiple context mode was added. |

## Examples

The following example shows how to configure a route map for OSPF routing:

```
ciscoasa(config)# route-map maptag1 permit 8
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# match metric 5
ciscoasa(config-route-map)# set metric-type type-2
ciscoasa(config-route-map)# show route-map
route-map maptag1 permit 8
  set metric 5
  set metric-type type-2
  match metric 5
ciscoasa(config-route-map)# exit
ciscoasa(config)#
```

## Related Commands

| Command | Description |
|---|---|
| **match interface** | Distributes any routes that have their next hop out one of the interfaces specified, |

| Command | Description |
|---|---|
| **route-map** | Defines the conditions for redistributing routes from one routing protocol into another. |
| **set metric** | Specifies the metric value in the destination routing protocol for a route map. |

# set metric-type internal

To set the Multi Exit Discriminator (MED) value on prefixes advertised to external BGP (eBGP) neighbors to match the Interior Gateway Protocol (IGP) metric of the next hop, use the set metric-type internal command in route-map configuration mode. To return to the default, use the no form of this command.

**set metric-type internal**
**no set metric-type internal**

## Syntax Description

This command has no arguments or keywords.

## 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 |
| Route-map configuration | • Yes | — | • Yes | • Yes | — |

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | We added this command. |

## Usage Guidelines

This command will cause BGP to advertise a MED value that corresponds to the IGP metric associated with the next hop of the route. This command applies to generated, internal BGP (iBGP)-, and eBGP-derived routes.

If this command is used, multiple BGP speakers in a common autonomous system can advertise different MED values for a particular prefix. Also, note that if the IGP metric changes, BGP will readvertise the route every 10 minutes.

Use the route-map global configuration command and the match and set route-map configuration commands to define the conditions for redistributing routes from one routing protocol into another. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which redistribution is allowed for the current route-map command. The set commands specify the set actions—the particular redistribution actions to perform if the criteria enforced by the match commands are met. The no route-map command deletes the route map.

The set route-map configuration commands specify the redistribution set actions to be performed when all of the match criteria of the route map are met. When all match criteria are met, all set actions are performed.

**Note** This command is not supported for redistributing routes into the Border Gateway Protocol (BGP).

## Examples

In the following example, the MED value for all the advertised routes to neighbor 172.16.2.3 is set to the corresponding IGP metric of the next hop:

```
ciscoasa(config)# router bgp 109
ciscoasa(config-router)# address-family ipv4
ciscoasa(config-router-af)# network 172.16.0.0
ciscoasa(config-router-af)# neighbor 172.16.2.3 remote-as 200
ciscoasa(config-router-af)# neighbor 172.16.2.3 route-map setMED out
ciscoasa(config-route-map)# route-map setMED permit 10
ciscoasa(config-route-map)# match as-path as-path-acl
ciscoasa(config-route-map)# set metric-type internal
ciscoasa(config-route-map)# ip as-path access-list as-path-acl permit .*
```

# set origin

To set the BGP origin code, use the set origin command in route-map configuration mode. To delete an entry, use the no form of this command.

**set origin** { **igp | egp** *autonomous-system-number* | **incomplete** }
**no set origin** { **igp | egp** *autonomous-system-number* | **incomplete** }

## Syntax Description

| | |
|---|---|
| autonomous-system-number | Number of a remote autonomous system number. The range of values for this argument is any valid autonomous system number from 1 to 65535. |
| egp | Local External Gateway Protocol (EGP) system. |
| igp | Remote Interior Gateway Protocol (IGP) system. |
| incomplete | Unknown heritage. |

## Command Default

The origin of the route is based on the path information of the route in the main IP routing table.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

You must have a match clause (even if it points to a “permit everything” list) if you want to set the origin of a route. Use this command to set a specific origin when a route is redistributed into BGP. When routes are redistributed, the origin is usually recorded as incomplete, identified with a ? in the BGP table.

Use the route-map global configuration command, and the match and set route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each route-map command has a list of match and set commands associated with it. The match commands specify the match criteria—the conditions under which redistribution is allowed for the current route-map command. The set commands specify the set actions—the particular redistribution actions to perform if the criteria enforced by the match commands are met. The no route-map command deletes the route map.

The set route-map configuration commands specify the redistribution set actions to be performed when all of the match criteria of a route map are met. When all match criteria are met, all set actions are performed.

## Examples

The following example sets the origin of routes that pass the route map to IGP:

```
ciscoasa(config-route-map)# route-map set_origin
ciscoasa(config-route-map)# match as-path 10
ciscoasa(config-route-map)# set origin igp
```

# set pfs

To specify the PFS group for the IPsec profile, use the set pfs command in the IPsec profile configuration mode. Use the no form of this command to remove the PFS group.

**set pfs** *Diffie-Hellman group* [ **group14** ]
**no set pfs** *Diffie-Hellman group* [ **group14** ]

## Syntax Description

| | |
|---|---|
| *Diffie-Hellman group* | Specifies the name of the *Diffie-Hellman group (dh group).* |
| group14 | Specifies that IPsec should use the 2048-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange. |

## 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 |
| IPsec profile configuration | • Yes | • No | • Yes | • No | — |

## Command History

| Release | Modification |
|---|---|
| 9.16(1) | Support was added for the group31 command option. |
| 9.15(1) | Support was removed for the group2 and group5 command options. |
| 9.13(1) | Added support for Group 14. The group2 and group5 command options was deprecated and will removed in the later release. |
| 9.7(1) | We introduced this command. |

## Examples

The following example sets group14 as the pfs:

```
ciscoasa(config)# crypto ipsec profile VTIipsec
ciscoasa(config-ipsec-profile)# set pfs group14
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ipsec profile** | Creates a new IPsec profile. |
| **responder-only** | Sets the VTI tunnel interface to responder only mode. |

| Command | Description |
|---|---|
| **set ikev1 transform-set** | Specifies the IKEv1 transform set to be used in the IPsec profile configuration. |
| **set security-association lifetime** | Specifies the duration of security association in the IPsec profile configuration. This is specified in kilobytes or seconds, or both. |
| set trustpoint | Specifies a trustpoint that defines the certificate to be used while initiating a VTI tunnel connection. |

# set security-association lifetime

To specify the duration of security association in the IPsec profile configuration, use the **set security-association lifetime** command in the IPsec profile configuration mode. This is specified in kilobytes or seconds, or both. Use the **no** form of this command to remove the security association lifetime configuration.

**set security-association lifetime** { **seconds** *number* | **kilobytes** { *number* | **unlimited** } }
**no set security-association lifetime** { **seconds** *number* | **kilobytes** { *number* | **unlimited** } }

## Syntax Description

| | |
|---|---|
| **kilobytes** { *number* \| **unlimited** } | Specifies the volume of traffic (in kilobytes) that can pass between peers using a given security association before that security association expires. The range is 10 to 2147483647 kbytes. The global default is 4,608,000 kilobytes.<br>This setting does not apply to remote access VPN connections. It applies to site-to-site VPN only. |
| **seconds** *number* | Specifies the number of seconds a security association will live before it expires. The range is 120 to 214783647 seconds. The global default is 28,800 seconds (eight hours).<br>This setting applies to both remote access and site-to-site VPN. |

## 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 |
| IPsec profile configuration | • Yes | • No | • Yes | • No | — |

## Command History

| Release | Modification |
|---|---|
| 9.7(1) | We introduced this command. |

## Usage Guidelines

The crypto map's security associations are negotiated according to the global lifetimes.

IPsec security associations use shared secret keys. These keys and their security associations time out together.

Assuming that the particular crypto map entry has lifetime values configured, when the ASA requests new security associations during security association negotiation, it specifies its crypto map lifetime values in the request to the peer; it uses these values as the lifetime of the new security associations. When the ASA receives a negotiation request from the peer, it uses the smaller of the lifetime values proposed by the peer or the locally configured lifetime values as the lifetime of the new security associations.

For site-to-site VPN connections, there are two lifetimes: a “timed” lifetime and a “traffic-volume” lifetime. The security association expires after the first of these lifetimes is reached. For remote access VPN sessions, only the timed lifetime applies.

**Note** The ASA lets you change crypto map, dynamic map, and IPsec settings on-the-fly. If you do so, the ASA brings down only the connections affected by the change. If you change an existing access list associated with a crypto map, specifically by deleting an entry within the access list, the result is that only the associated connection is brought down. Connections based on other entries in the access list are not affected.

## Examples

The following example sets the security association lifetime values:

```
ciscoasa(config)# crypto ipsec profile VTIipsec
ciscoasa(config-ipsec-profile)# set security-association lifetime seconds 120 kilobytes
10000
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ipsec profile** | Creates a new IPsec profile. |
| **responder-only** | Sets the VTI tunnel interface to responder only mode. |
| **set ikev1 transform-set** | Specifies the IKEv1 transform set to be used in the IPsec profile configuration. |
| **set pfs** | Specifies the PFS group to be used in the IPsec profile configuration. |
| **set trustpoint** | Specifies a trustpoint that defines the certificate to be used while initiating a VTI tunnel connection. |

# set trustpoint

To specify a trustpoint that defines the certificate to be used while initiating a VTI tunnel connection, use the set trustpoint command in the IPsec profile configuration mode. Use the no form of this command to remove the trustpoint configuration.

**set trustpoint** *name* **chain**
**no set trustpoint** *name* **chain**

## Syntax Description

| | |
|---|---|
| *name* | Specifies the name of the trustpoint. |
| chain | Enables the sending of certificate chain. |

## 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 |
| IPsec profile configuration | • Yes | • No | • Yes | • No | — |

## Command History

| Release | Modification |
|---|---|
| 9.8(1) | We introduced this command. |

## Examples

The following example sets the security association lifetime values:

```
ciscoasa(config)# crypto ipsec profile VTIipsec
ciscoasa(config-ipsec-profile)# set trustpoint TPVTI chain
```

## Related Commands

| Command | Description |
|---|---|
| **crypto ipsec profile** | Creates a new IPsec profile. |
| **responder-only** | Sets the VTI tunnel interface to responder only mode. |
| **set ikev1 transform-set** | Specifies the IKEv1 transform set to be used in the IPsec profile configuration. |
| **set pfs** | Specifies the PFS group to be used in the IPsec profile configuration. |

# setup

To configure a minimal configuration for the ASA using interactive prompts, enter the **setup** command in global configuration mode.

**setup**

## 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 | • Yes |

## Command History

| Release | Modification |
|---|---|
| 7.0(1) | This command was added. |
| 8.4(1) | In routed mode for the ASA 5510 and higher, the interface configured is now the Management *slot* /*port* interface, and not the “inside” interface. For the ASA 5505, the interface configured is the VLAN 1 interface, not “inside”. |
| 9.0(1) | The default configuration prompt was changed, and Ctrl + Z to exit the setup process was enabled. |

## Usage Guidelines

The setup prompt automatically appears at boot time if there is no startup configuration in flash memory.

The **setup** command walks you through minimal configuration to establish ASDM connectivity. This command is designed for a unit that has either no configuration or a partial configuration. If your model supports a factory default configuration, we recommend using the factory default configuration instead of the **setup** command (to restore the default configuration, use the **configure factory-default** command).

The **setup** command requires an already-named interface called “management.”

When you enter the **setup** command, you are asked for the information in XREF. If there is already a configuration for the listed parameter, it appears in brackets, so you can either accept it as the default or override it by entering a new value. The exact prompts available may differ per model. The system **setup** command includes a subset of these prompts.

*Table 1: Setup Prompts*

| Prompt | Description |
|---|---|
| `Pre-configure Firewall now through interactive prompts [yes]?` | Enter **yes** or **no**. If you enter **yes**, the setup continues. If **no**, the setup stops and the global configuration prompt (ciscoasa(config)#) appears. |
| `Firewall Mode [Routed]:` | Enter **routed** or **transparent**. |
| `Enable password:` | Enter an enable password. (The password must have at least three characters.) |
| `Allow password recovery [yes]?` | Enter **yes** or **no**. |
| `Clock (UTC):` | You cannot enter anything in this field. The UTC time is used by default. |
| `Year:` | Enter the year using four digits, for example, 2005. The year range is 1993 to 2035. |
| `Month:` | Enter the month using the first three characters of its name, for example, **Sep** for September. |
| `Day:` | Enter the day of the month, from 1 to 31. |
| `Time:` | Enter the hour, minutes, and seconds in 24-hour time format, for example, enter **20:54:44** for 8:54 p.m and 44 seconds. |
| `Host name:` | Enter the hostname that you want to display in the command line prompt. |
| `Domain name:` | Enter the domain name of the network on which the ASA runs. |
| `IP address of host running Device Manager:` | Enter the IP address of the host that needs to access ASDM. |
| `Use this configuration and save to flash (yes)?` | Enter **yes** or **no**. If you enter **yes**, the inside interface is enabled and the requested configuration is written to the Flash partition.<br>If you enter **no**, the setup prompt repeats, beginning with the first question:<br>`Pre-configure Firewall now through interactive prompts [yes]?`<br>Enter **Ctrl + Z** to exit the setup or **yes** to repeat the prompt. |

## Examples

The following example shows how to complete the **setup** command:

```
ciscoasa(config)# setup
Pre-configure Firewall now through interactive prompts [yes]? yes
```

```
Firewall Mode [Routed]: routed
Enable password [<use current password>]: writer
Allow password recovery [yes]? yes
Clock (UTC):
   Year: 2005
   Month: Nov
   Day: 15
   Time: 10:0:0
Inside IP address: 192.168.1.1
Inside network mask: 255.255.255.0
Host name: tech_pubs
Domain name: example.com
IP address of host running Device Manager: 10.1.1.1
The following configuration will be used:
Enable password: writer
Allow password recovery: yes
Clock (UTC): 20:54:44 Sep 17 2005
Firewall Mode: Routed
Inside IP address: 192.168.1.1
Inside network mask: 255.255.255.0
Host name: tech_pubs
Domain name: example.com
IP address of host running Device Manager: 10.1.1.1
Use this configuration and write to flash? yes
```

## Related Commands

| Command | Description |
|---|---|
| **configure factory-default** | Restores the default configuration. |

# set weight

To specify the BGP weight for the routing table, use the set weight command in route-map configuration mode. To delete an entry, use the no form of this command.

**set weight** *number*
**no set weight** *number*

## Syntax Description

| | |
|---|---|
| number | Weight value. It can be an integer ranging from 0 to 65535. |

## Command Default

The weight is not changed by the specified route map.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

The implemented weight is based on the first matched autonomous system path. Weights indicated when an autonomous system path is matched override the weights assigned by global neighbor commands. In other words, the weights assigned with the set weight route-map configuration command override the weights assigned using the neighbor weight command.

## Examples

The following example sets the BGP weight for the routes matching the autonomous system path access list to 200:

```
ciscoasa(config-route-map)# route-map set-weight
ciscoasa(config-route-map)# match as-path as_path_acl
iscoasa(config-route-map)# set weight 200
```

# sfr

To redirect traffic to the ASA FirePOWER module, use the **sfr** command in class configuration mode. To remove the redirect, use the **no** form of this command.

**sfr** { **fail-close | fail-open** } [ **monitor-only** ]
**no sfr** { **fail-close | fail-open** } [ **monitor-only** ]

## Syntax Description

| | |
|---|---|
| **fail-close** | Sets the ASA to block the traffic if the module is unavailable. |
| **fail-open** | Sets the ASA to allow the traffic through, applying ASA policies only, if the module is unavailable. |
| **monitor-only** | Sends a read-only copy of traffic to the module, i.e. passive mode. If you do not include the keyword, the traffic is sent in inline mode. |

## Command Default

No default behavior or values.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 9.2(1) | This command was added. |

## Usage Guidelines

You can access the class configuration mode by first entering the policy-map command.

Before or after you configure the **sfr** command on the ASA, configure the security policy on the module using Secure Firewall Management Center (formerly Firepower Management Center).

To configure the **sfr** command, you must first configure the **class-map** command, **policy-map** command, and the **class** command.

**Traffic Flow**

The ASA FirePOWER module runs a separate application from the ASA. It is, however, integrated into the ASA traffic flow. When you apply the **sfr** command for a class of traffic on the ASA, traffic flows through the ASA and the module in the following way:

**1.** Traffic enters the ASA.

**2.** Incoming VPN traffic is decrypted.

**3.** Firewall policies are applied.

**4.** Traffic is sent to the ASA FirePOWER module over the backplane.

**5.** The module applies its security policy to the traffic and takes appropriate actions.

**6.** In inline mode, valid traffic is sent back to the ASA over the backplane; the ASA FirePOWER module might block some traffic according to its security policy, and that traffic is not passed on. In passive mode, no traffic is returned, and the module cannot block traffic.

**7.** Outgoing VPN traffic is encrypted.

**8.** Traffic exits the ASA.

**Compatibility with ASA Features**

The ASA includes many advanced application inspection features, including HTTP inspection. However, the ASA FirePOWER module provides more advanced HTTP inspection than the ASA provides, as well as additional features for other applications, including monitoring and controlling application usage.

To take full advantage of the ASA FirePOWER module features, see the following guidelines for traffic that you send to the ASA FirePOWER module:

Do not configure ASA inspection on HTTP traffic.

- Do not configure Cloud Web Security (ScanSafe) inspection. If you configure both ASA FirePOWER inspection and Cloud Web Security inspection for the same traffic, the ASA only performs ASA FirePOWER inspection.
- Other application inspections on the ASA are compatible with the ASA FirePOWER module, including the default inspections.
- Do not enable the Mobile User Security (MUS) server; it is not compatible with the ASA FirePOWER module.
- If you enable failover, when the ASA fails over, any existing ASA FirePOWER flows are transferred to the new ASA. The ASA FirePOWER module in the new ASA begins inspecting the traffic from that point forward; old inspection states are not transferred.

**Monitor-Only Mode**

The traffic flow in monitor-only mode is the same as it is for inline mode. The only difference is that the ASA FirePOWER module does not pass traffic back to the ASA. Instead, the module applies the security policy to the traffic and lets you know what it would have done if it were operating in inline mode, e.g. traffic might be marked “would have dropped” in events. You can use this information for traffic analysis and to help you decide if inline mode is desirable.

**Note** You cannot configure both monitor-only mode and normal inline mode at the same time on the ASA. Only one type of security policy is allowed. In multiple context mode, you cannot configure monitor-only mode for some contexts, and regular inline mode for others.

## Examples

The following example diverts all HTTP traffic to the ASA FirePOWER module, and blocks all HTTP traffic if the module fails for any reason:

```
ciscoasa(config)# access-list ASASFR permit tcp any any eq port 80
ciscoasa(config)# class-map my-sfr-class
ciscoasa(config-cmap)# match access-list ASASFR
```

```
ciscoasa(config-cmap)# policy-map my-sfr-policy
ciscoasa(config-pmap)# class my-sfr-class
ciscoasa(config-pmap-c)# sfr fail-close
ciscoasa(config-pmap-c)# service-policy my-cx-policy global
```

The following example diverts all IP traffic destined for the 10.1.1.0 network and the 10.2.1.0 network to the ASA FirePOWER module, and allows all traffic through if the module fails for any reason.

```
ciscoasa(config)# access-list my-sfr-acl permit ip any 10.1.1.0 255.255.255.0
ciscoasa(config)# access-list my-sfr-acl2 permit ip any 10.2.1.0 255.255.255.0
ciscoasa(config)# class-map my-sfr-class
ciscoasa(config-cmap)# match access-list my-sfr-acl
ciscoasa(config)# class-map my-sfr-class2
ciscoasa(config-cmap)# match access-list my-sfr-acl2
ciscoasa(config-cmap)# policy-map my-sfr-policy
ciscoasa(config-pmap)# class my-sfr-class
ciscoasa(config-pmap-c)# sfr fail-open
ciscoasa(config-pmap)# class my-sfr-class2
ciscoasa(config-pmap-c)# sfr fail-open
ciscoasa(config-pmap-c)# service-policy my-sfr-policy interface outside
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Specifies a class map to use for traffic classification. |
| **class-map** | Identifies traffic for use in a policy map. |
| **cxsc** | Redirects traffic to the ASA CX module. |
| **hw-module module reload** | Reloads the module. |
| **hw-module module reset** | Performs a reset and then reloads the module. |
| **hw-module module shutdown** | Shuts down the module. |
| **policy-map** | Configures a policy; that is, an association of a traffic class and one or more actions. |
| **show asp table classify domain sfr** | Shows the NP rules created to send traffic to the ASA FirePOWER module. |
| **show module** | Shows the module status. |
| **show running-config policy-map** | Displays all current policy map configurations. |
| **show service-policy** | Shows service policy statistics. |
| sw-module module sfr reload | Reloads the software module. |
| sw-module module sfr reset | Resets the software module. |
| sw-module module sfr recover | Installs the software module boot image. |
| sw-module module sfr shutdown | Shuts down the software module. |

# shape

To enable QoS traffic shaping, use the **shape** command in class configuration mode. If you have a device that transmits packets at a high speed, such as a ASA with Fast Ethernet, and it is connected to a low speed device such as a cable modem, then the cable modem is a bottleneck at which packets are frequently dropped. To manage networks with differing line speeds, you can configure the ASA to transmit packets at a fixed slower rate, called *traffic shaping* . To remove this configuration, use the **no** form of this command.

**Note** Traffic shaping is only supported on the ASA 5505, 5510, 5520, 5540, and 5550. Multi-core models (such as the ASA 5500-X) do not support shaping.

**shape average** *rate* [ *burst_size* ]
**no shape average** *rate* [ *burst_size* ]

## Syntax Description

| | |
|---|---|
| **average** *rate* | Sets the average rate of traffic in bits per second over a given fixed time period, between 64000 and 154400000. Specify a value that is a multiple of 8000. See the "Usage Guidelines" section for more information about how the time period is calculated. |
| *burst_size* | Sets the average burst size in bits that can be transmitted over a given fixed time period, between 2048 and 154400000. Specify a value that is a multiple of 128. If you do not specify the *burst_size* , the default value is equivalent to 4-milliseconds of traffic at the specified average rate. For example, if the average rate is 1000000 bits per second, 4 ms worth = 1000000 * 4/1000 = 4000. |

## Command Default

If you do not specify the *burst_size* , the default value is equivalent to 4-milliseconds of traffic at the specified average rate. For example, if the average rate is 1000000 bits per second, 4 ms worth = 1000000 * 4/1000 = 4000.

## Command Modes

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

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

## Command History

| Release | Modification |
|---|---|
| 7.2(4)/8.0(4) | This command was added. |

## Usage Guidelines

To enable traffic shaping, use the Modular Policy Framework:

1. **policy-map** —Identify the actions associated with the **class-default** class map.

**a. class class-default** —Identify the **class-default** class map on which you want to perform actions.

**b. shape** —Apply traffic shaping to the class map.

**c.** (Optional) **service-policy** —Call a different policy map in which you configured the **priority** command so you can apply priority queueing to a subset of shaped traffic.

**2. service-policy** —Assigns the policy map to an interface or globally.

**Traffic Shaping Overview**

Traffic shaping is used to match device and link speeds, thereby controlling packet loss, variable delay, and link saturation, which can cause jitter and delay.

- Traffic shaping must be applied to all outgoing traffic on a physical interface or in the case of the ASA 5505, on a VLAN. You cannot configure traffic shaping for specific types of traffic.
- Traffic shaping is implemented when packets are ready to be transmitted on an interface, so the rate calculation is performed based on the actual size of a packet to be transmitted, including all the possible overhead such as the IPsec header and L2 header.
- The shaped traffic includes both through-the-box and from-the-box traffic.
- The shape rate calculation is based on the standard token bucket algorithm. The token bucket size is twice the burst size value. See the CLI configuration guide for more information about the token bucket.
- When bursty traffic exceeds the specified shape rate, packets are queued and transmitted later. Following are some characteristics regarding the shape queue (for information about hierarchical priority queuing, see the **priority** command):
- The queue size is calculated based on the shape rate. The queue can hold the equivalent of 200-milliseconds worth of shape rate traffic, assuming a 1500-byte packet. The minimum queue size is 64.
- When the queue limit is reached, packets are tail-dropped.
- Certain critical keep-alive packets such as OSPF Hello packets are never dropped.
- The time interval is derived by *time_interval* = *burst_size* / *average_rate* . The larger the time interval is, the burstier the shaped traffic might be, and the longer the link might be idle. The effect can be best understood using the following exaggerated example:

Average Rate = 1000000

Burst Size = 1000000

In the above example, the time interval is 1 second, which means, 1 Mbps of traffic can be bursted out within the first 10 milliseconds of the 1-second interval on a 100 Mbps FE link and leave the remaining 990 milliseconds idle without being able to send any packets until the next time interval. So if there is delay-sensitive traffic such as voice traffic, the Burst Size should be reduced compared to the average rate so the time interval is reduced.

**How QoS Fea tures Interact**

You can configure each of the QoS features alone if desired for the ASA. Often, though, you configure multiple QoS features on the ASA so you can prioritize some traffic, for example, and prevent other traffic from causing bandwidth problems.

See the following supported feature combinations per interface:

- Standard priority queuing (for specific traffic) + Policing (for the rest of the traffic).

You cannot configure priority queuing and policing for the same set of traffic.

- Traffic shaping (for all traffic on an interface) + Hierarchical priority queuing (for a subset of traffic).

You cannot configure traffic shaping and standard priority queuing for the same interface; only hierarchical priority queuing is allowed. For example, if you configure standard priority queuing for the global policy, and then configure traffic shaping for a specific interface, the feature you configured last is rejected because the global policy overlaps the interface policy.

Typically, if you enable traffic shaping, you do not also enable policing for the same traffic, although the ASA does not restrict you from configuring this.

## Examples

The following example enables traffic shaping for all traffic on the outside interface, and prioritizes traffic within VPN tunnel-grp1 with the DSCP bit set to ef:

```
ciscoasa
(config)#
class-map TG1-voice
ciscoasa
(config-cmap)#
match tunnel-group tunnel-grp1
ciscoasa
(config-cmap)#
match dscp ef
ciscoasa(config)# policy-map priority-sub-policy
ciscoasa(config-pmap)# class
            TG1-voice
ciscoasa(config-pmap-c)# priority
ciscoasa(config-pmap-c)# policy-map shape_policy
ciscoasa(config-pmap)# class
            class-default
ciscoasa(config-pmap-c)# shape
ciscoasa(config-pmap-c)# service-policy priority-sub-policy
ciscoasa
(config-pmap-c)#
service-policy shape_policy
 interface outside
```

## Related Commands

| Command | Description |
|---|---|
| **class** | Identifies the class map on which you want to perform actions in a policy map. |
| **police** | Enables QoS policing. |
| **policy-map** | Identifies actions to apply to traffic in a service policy. |
| **priority** | Enables QoS priority queuing. |
| service-policy (class) | Applies a hierarchical policy map. |
| service-policy (global) | Applies a service policy to interface(s). |
| **show service-policy** | Shows QoS statistics. |

# share-ratio

To configure the port ratio, which determines how many ports are in the port pool in the basic mapping rule in a Mapping Address and Port (MAP) domain, use the **share-ratio** command in MAP domain basic mapping rule configuration mode. Use the **no** form of this command to remove the ratio.

**share-ratio***number*
**no share-ratio** *number*

## Syntax Description

| | |
|---|---|
| *number* | The number of ports that should be in the pool. The number must be a power of 2, from 1-65536, such as 1, 2, 4, 8, and so forth. |

## 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. |