CHAPTER 9

crypto ca authenticate through crypto map set trustpoint Commands

crypto ca authenticate

To install and authenticate the CA certificates associated with a trustpoint, use the **crypto ca authenticate** command in global configuration mode. To remove the CA certificate, use the **no** form of this command.

crypto ca authenticate *trustpoint* [**fingerprint** *hexvalue*] [**nointeractive]**

no crypto ca authenticate *trustpoint*

Syntax Description

fingerprint	Specifies a hash value consisting of alphanumeric characters the FWSM uses to authenticate the CA certificate. If a fingerprint is provided, the FWSM compares it to the computed fingerprint of the CA certificate and accepts the certificate only if the two values match. If there is no fingerprint, the FWSM displays the computed fingerprint and asks whether to accept the certificate.
hexvalue	Identifies he hexadecimal value of the fingerprint.
nointeractive	Obtains the CA certificate for this trustpoint using no interactive mode; intended for use by the device manager only. In this case, if there is no fingerprint, the FWSM accepts the certificate without question.
trustpoint	Specifies the trustpoint from which to obtain the CA certificate. Maximum name length is 128 characters.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

If the trustpoint is configured for SCEP enrollment, the CA certificate is downloaded through SCEP. If not, the FWSM prompts you to paste the base-64 formatted CA certificate onto the terminal.

The invocations of this command do not become part of the running configuration.

Examples

In the following example, the FWSM requests the certificate of the CA. The CA sends its certificate and the FWSM prompts the administrator to verify the certificate of the CA by checking the CA certificate fingerprint. The FWSM administrator should verify the fingerprint value displayed against a known, correct value. If the fingerprint displayed by the FWSM matches the correct value, you should accept the certificate as valid.

```
hostname(config)# crypto ca authenticate myca
Certificate has the following attributes:
Fingerprint: 0123 4567 89AB CDEF 0123
Do you accept this certificate? [yes/no] y#
hostname(config)#
```

In the next example, the trustpoint tp9 is configured for terminal-based (manual) enrollment. In this case theFWSM prompts the administrator to paste the CA certificate to the terminal. After displaying the fingerprint of the certificate, the FWSM prompts the administrator to confirm that the certificate should be retained.

```
hostname(config)# crypto ca authenticate tp9
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

MIIDjjCCAvegAwIBAgIQejIaQ3SJRIBMHcvDdgOsKTANBgkqhkiG9w0BAQUFADBA
MQswCQYDVQQGEwJVUzELMAkGA1UECBMCTUExETAPBgNVBAcTCEZyYW5rbGluMREw
DwYDVQQDEwhCcmlhbnNDQTAeFw0wMjEwMTcxODE5MTJaFw0wNjEwMjQxOTU3MDha
MEAxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJNQTERMA8GA1UEBxMIRnJhbmtsaW4x
ETAPBgNVBAMTCEJyaWFuc0NBMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCd
jXEPvNnkZD1bKzahbTHuRot1T8KRUbCP5aWKfqViKJENzI2GnAheArazsAcc4Eaz
LDnpuyyqa0j5LA3MI577MoN1/nll018fbpqOf9eVDPJDkYTvtZ/X3vJgnEjTOWyz
T0pXxhdU1b/jgqVE74OvKBzU7A2yoQ2hMYzwVbGkewIDAQABo4IBhzCCAYMwEwYJ
KwYBBAGCNxQCBAYeBABDAEEwCwYDVR0PBAQDAgFGMA8GA1UdEwEB/wQFMAMBAf8w
HQYDVR0OBBYEFBHr3holowFDmniI3FBwKpSEucdtMIIBGwYDVR0fBIIBEjCCAQ4w
gcaggcOggcCGgb1sZGFwOi8vL0NOPUJyaWFuc0NBLENOPWJyaWFuLXcyay1zdnIs
Q049Q0RQLENOPVB1YmxpYyUyMEtleSUyMFNlcnZpY2VzLENOPVNlcnZpY2VzLENO
PUNvbmZpZ3VyYXRpb24sREM9YnJpYW5wZGMsREM9YmRzLERDPWNvbT9jZXJ0aWZp
Y2F0ZVJldm9jYXRpb25MaXN0P2Jhc2U/b2JqZWN0Y2xhc3M9Y1JMRGlzdHJpYnV0
aW9uUG9pbnQwQ6BBoD+GPWh0dHA6Ly9icmlhbi13Mmstc3ZyLmJyaWFucGRjLmJk
cy5jb20vQ2VydEVucm9sbC9CcmlhbnNDQS5jcmwwEAYJKwYBBAGCNxUBBAMCAQEw
DQYJKoZIhvcNAQEFBQADgYEAdLhc4Za3AbMjRq66xH1qJWxKUzd4nE9wOrhGgA1r
j4B/Hv2K1gUie34xGqu9OpwqvJgp/vCU12Ciykb1YdSDy/PxN4KtR9Xd1JDQMbu5
f20AYqCG5vpPWavCgmgTLcdwKa3ps1YSWGkhWmScHHSiGg1a3tevYVwhHNPA4mWo
7sQ=

Certificate has the following attributes:
Fingerprint: 21B598D5 4A81F3E5 0B24D12E 3F89C2E4
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
hostname(config)#
```

Related Commands

Command	Description
crypto ca enroll	Starts enrollment with a CA.
crypto ca import certificate	Installs a certificate received from a CA in response to a manual enrollment request. Also used to import PKS12 data to a trustpoint.
crypto ca trustpoint	Enters the trustpoint submode for the indicated trustpoint.

crypto ca certificate chain

To enter certificate chain configuration mode for the indicated trustpoint, use the **crypto ca certificate chain** command in global configuration mode. To return to global configuration mode, use the **no** form of this command or use the **exit** command.

crypto ca certificate chain *trustpoint*

[no] crypto ca certificate chain *trustpoint*

Syntax Description

trustpoint	Specifies the trustpoint for configuring the certificate chain.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Examples

The following example enters CA certificate chain submode for trustpoint central:

```
hostname<config># crypto ca certificate chain central
hostname<config-cert-chain>#
```

Related Commands

Command	Description
clear configure crypto ca trustpoint	Removes all trustpoints.

crypto ca certificate map

To enter CA certificate map mode, use the **crypto ca configuration map** command in global configuration mode. To remove a crypto CA configuration map rule, use the **no** form of the command.

crypto ca certificate map *sequence-number*

no crypto ca certificate map [*sequence-number*]

Syntax Description

sequence-number	Specifies a number for the certificate map rule you are creating. The range is 1 through 65535. You can use this number when creating a tunnel-group-map, which maps a tunnel group to a certificate map rule.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Executing this command places you in ca-certificate-map mode. Use this group of commands to maintain a prioritized list of certificate mapping rules. The sequence number orders the mapping rules.

Issuing this command places the FWSM in CA certificate map configuration mode where you can configure rules based on the certificate's issuer and subject distinguished names (DNs). The general form of these rules is as follows:

DN match-criteria match-value

DN is either *subject-name* or *issuer-name*. DNs are defined in the ITU-T X.509 standard. For a list of certificate fields, see Related Commands.

match-criteria comprise the following expressions or operators:

attr *tag*	Limits the comparison to a specific DN attribute, such as common name (CN).
co	Contains
eq	Equal

nc	Does not contain
ne	Not equal

The DN matching expressions are case insensitive.

Examples

The following example enters CA certificate map mode with a sequence number of 1 (rule # 1) and specifies that the common name(CN) attribute of the subject-name must match user1:

```
hostname(config)# crypto ca certificate map 1
hostname(ca-certificate-map)# subject-name attr cn eq user1
hostname(ca-certificate-map)#
```

The following example enters CA certificate map mode with a sequence number of 1 and specifies that the subject-name contain the value cisco anywhere within it:

```
hostname(config)# crypto ca certificate map 1
hostname(ca-certificate-map)# subject-name co cisco
hostname(ca-certificate-map)#
```

Related Commands+

Command	Description
issuer-name	Indicates that rule entry is applied to the issuer DN of the IPSec peer certificate.
subject-name (crypto ca certificate map)	Indicates that rule entry is applied to the subject DN of the IPSec peer certificate.
tunnel-group-map enable	Associates the certificate map entries created using the **crypto ca certificate map** command with tunnel groups.

crypto ca crl request

To request a CRL based on the configuration parameters of the specified trustpoint, use the **crypto ca crl request** command in Crypto ca trustpoint configuration mode.

crypto ca crl request *trustpoint*

Syntax Description

trustpoint	Specifies the trustpoint. Maximum number of characters is 128.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Invocations of this command do not become part of the running configuration.

Examples

The following example requests a CRL based on the trustpoint named central:

```
hostname(config)# crypto ca crl request central
hostname(config)#
```

Related Commands

Command	Description
crl configure	Enters crl configuration mode.

crypto ca enroll

To start the enrollment process with the CA, use the **crypto ca enroll** command in global configuration mode. For this command to execute successfully, the trustpoint must have been configured correctly.

crypto ca enroll *trustpoint* [**noconfirm**]

Syntax Description

noconfirm	(Optional) Suppresses all prompts. Enrollment options that might have been prompted for must be pre-configured in the trustpoint. This option is for use in scripts, ASDM, or other such non-interactive needs.
trustpoint	Specifies the name of the trustpoint to enroll with. Maximum number of characters is 128.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

When the trustpoint is configured for SCEP enrollment, the FWSM displays a CLI prompt immediately and displays status messages to the console asynchronously. When the trustpoint is configured for manual enrollment, the FWSM writes a base-64-encoded PKCS10 certification request to the console and then displays the CLI prompt.

This command generates interactive prompts that vary depending on the configured state of the referenced trustpoint.

Examples

The following example shows how to enroll for an identity certificate with trustpoint tp1 using SCEP enrollment. The FWSM prompts for information not stored in the trustpoint configuration.

```
hostname(config)# crypto ca enroll tp1
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally provide this
% password to the CA Administrator in order to revoke your certificate.
% For security reasons your password will not be saved in the configuration.
% Please make a note of it.
Password:
```

```
Re-enter password:
% The fully-qualified domain name in the certificate will be: xyz.example.com
% The subject name in the certificate will be: xyz.example.com
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Request certificate from CA [yes/no]: yes
% Certificate request sent to Certificate authority.
% The certificate request fingerprint will be displayed.
% The 'show crypto ca certificate' command will also show the fingerprint.

hostname(config)#
```

The following example shows manual enrollment of a CA certificate.

```
hostname(config)# crypto ca enroll tp1
% Start certificate enrollment ..
% The fully-qualified domain name in the certificate will be: xyz.example.com
% The subject name in the certificate will be: wb-2600-3.example.com
if serial number not set in trustpoint, prompt:
% Include the router serial number in the subject name? [yes/no]: no
If ip-address not configured in trustpoint:
% Include an IP address in the subject name? [no]: yes
Enter Interface name or IP Address[]: 1.2.3.4
Display Certificate Request to terminal? [yes/no]: y
Certificate Request follows:
MIIBFTCBwAIBADA6MTgwFAYJKoZIhvcNAQkIEwcxLjIuMy40MCAGCSqGSIb3DQEJ
AhYTd2ItMjYwMC0zLmNpc2NvLmNvbTBcMA0GCSqGSIb3DQEBAQUAA0sAMEgCQQDT
IdvHa4D5wXZ+40sKQV7Uek1E+CC6hm/LRN3p5ULW1KF6bxhA3Q5CQfh4jDxobn+A
Y8GoeceulS2Zb+mvgNvjAgMBAAGgITAfBgkqhkiG9w0BCQ4xEjAQMA4GA1UdDwEB
/wQEAwIFoDANBgkqhkiG9w0BAQQFAANBACDhnrEGBVtltG7hp8x6Wz/dgY+ouWcA
lzy7QpdGhb1du2P81RYn+8pWRA43cikXMTeM4ykEkZhLjDUgv9t+R9c=

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]: no
hostname(config)#
```

Related Commands

Command	Description
crypto ca authenticate	Obtains the CA certificate for this trustpoint.
crypto ca import pkcs12	Installs a certificate received from a CA in response to a manual enrollment request. Also used to import PKS12 data to a trustpoint.
crypto ca trustpoint	Enters the trustpoint submode for the indicated trustpoint.

crypto ca export

To export in PKCS12 format the keys and certificates associated with a trustpoint configuration, use the **crypto ca export** command in global configuration mode.

crypto ca export *trustpoint* **pkcs12** *passphrase*

Syntax Description

passphrase	Specifies the passphrase used to encrypt the PKCS12 file for export.
pkcs12	Specifies the public key cryptography standard to use in exporting the trustpoint configuration.
trustpoint	Specifies the name of the trustpoint whose certificate and keys are to be exported. When you export, if the trustpoint uses RSA keys, the exported key pair is assigned the same name as the trustpoint.

Defaults

This command has no default values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Invocations of this command do not become part of the active configuration. The PKCS12 data is written to the terminal.

Examples

The following example exports PKCS12 data for trustpoint central using xxyyzz as the passcode:

```
hostname(config)# crypto ca export central pkcs12 xxyyzz

Exported pkcs12 follows:

[ PKCS12 data omitted ]

---End - This line not part of the pkcs12---
```

Related Commands

Command	Description
crypto ca import pkcs12	Installs a certificate received from a CA in response to a manual enrollment request. Also used to import PKS12 data to a trustpoint.
crypto ca authenticate	Obtains the CA certificate for this trustpoint.
crypto ca enroll	Starts enrollment with a CA.
crypto ca trustpoint	Enters the trustpoint submode for the indicated trustpoint.

crypto ca import

To install a certificate received from a CA in response to a manual enrollment request or to import the certificate and key pair for a trustpoint using PKCS12 data, use the **crypto ca import** command in global configuration mode. The FWSM prompts you to paste the text to the terminal in base 64 format.

crypto ca import *trustpoint* **certificate** [**nointeractive**]

crypto ca import *trustpoint* **pkcs12** *passphrase* [**nointeractive**]

Syntax Description

certificate	Tells the FWSM to import a certificate from the CA represented by the trustpoint.
nointeractive	(Optional) Imports a certificate using nointeractive mode. This suppresses all prompts. This option for use in scripts, ASDM, or other such non-interactive needs.
passphrase	Specifies the passphrase used to decrypt the PKCS12 data.
pkcs12	Tells the FWSM to import a certificate and key pair for a trustpoint, using PKCS12 format.
trustpoint	Specifies the trustpoint with which to associate the import action. Maximum number of characters is 128. If you import PKCS12 data and the trustpoint uses RSA keys, the imported key pair is assigned the same name as the trustpoint.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Examples

The following example manually imports a certificate for the trustpoint Main:

```
hostname(config)# crypto ca import Main certificate
% The fully-qualified domain name in the certificate will be:
securityappliance.example.com

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself
[ certificate data omitted ]
```

```
quit
INFO: Certificate successfully imported
```

The following example manually imports PKCS12 data to trustpoint central:

```
hostname(config)# crypto ca import central pkcs12

Enter the base 64 encoded pkcs12.
End with a blank line or the word "quit" on a line by itself:
[ PKCS12 data omitted ]
quit
INFO: Import PKCS12 operation completed successfully
```

Related Commands

Command	Description
crypto ca export	Exports a trustpoint certificate and key pair in PKCS12 format.
crypto ca authenticate	Obtains the CA certificate for a trustpoint.
crypto ca enroll	Starts enrollment with a CA.
crypto ca trustpoint	Enters the trustpoint submode for the indicated trustpoint.

crypto ca trustpoint

To add a trustpoint and enter trustpoint configuration mode, use the **crypto ca trustpoint** command in global configuration mode. To remove the specified trustpoint, use the **no** form of this command.

crypto ca trustpoint *trustpoint-name*

no crypto ca trustpoint *trustpoint-name* [**noconfirm**]

Syntax Description

noconfirm	(Optional) Suppresses all interactive prompting.
trustpoint- name	Identifies the name of the trustpoint to manage. The maximum name length is 128 characters.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	Support for this command was introduced.

Usage Guidelines

This command manages trustpoint information. A trustpoint represents a CA identity and possibly a device identity, based on a certificate issued by the CA. The trustpoint commands control CA-specific configuration parameters which specify how the FWSM obtains the CA certificate, how the FWSM obtains its certificate from the CA, and the authentication policies for user certificates issued by the CA.

A trustpoint represents a CA identity and possibly a device identity, based on a certificate issued by the CA. The trustpoint commands control CA-specific configuration parameters which specify how the FWSM obtains the CA certificate, how the FWSM obtains its certificate from the CA, and the authentication policies for user certificates issued by the CA.

Examples

The following example enters CA trustpoint mode for managing a trustpoint named central:

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

Related Commands

Command	Description
clear configure crypto ca trustpoint	Removes all trustpoints.
crypto ca authenticate	Obtains the CA certificate for this trustpoint.
crypto ca certificate map	Enters crypto CA certificate map mode. Defines certificate-based ACLs.
crypto ca crl request	Requests a CRL based on configuration parameters of specified trustpoint.
crypto ca import	Installs a certificate received from a CA in response to a manual enrollment request. Also used to import PKS12 data to a trustpoint.

crypto dynamic-map match address

To define a dynamic crypto map entry, use the **crypto dynamic-map match address** command in global configuration mode. To remove the access list from a crypto map entry, use the **no** form of this command. See the **crypto map match address** command for additional information about this command.

crypto dynamic-map *dynamic-map-name dynamic-seq-num* **match address** *acl_name*

no crypto dynamic-map *dynamic-map-name dynamic-seq-num* **match address** *acl_name*

Syntax Description

acl-name	Identifies the access list to be matched for the dynamic crypto map entry.
dynamic-map-name	Specifies the name of the dynamic crypto map set.
dynamic-seq-num	Specifies the sequence number that corresponds to the dynamic crypto map entry.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was changed from **crypto dynamic-map**.

Examples

The following example shows the use of the crypto dynamic-map command to match address of an access list named aclist1:

```
hostname(config)# crypto dynamic-map mymap 10 match address aclist1
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto dynamic-map	Clears all configuration for all the dynamic crypto maps.
show running-config crypto dynamic-map	Displays all configuration for all the dynamic crypto maps.

crypto dynamic-map set peer

To define a dynamic crypto map entry, use the **crypto dynamic-map set peer** command in global configuration mode.To remove the access list from a crypto map entry, use the **no** form of this command. See the **crypto map set peer** command for additional information about this command.

crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set peer** *ip_address* | *hostname*

no crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set peer** *ip_address* | *hostname*

Syntax Description

dynamic-map-name	Specifies the name of the dynamic crypto map set.
dynamic-seq-num	Specifies the sequence number that corresponds to the dynamic crypto map entry.
hostname	Identifies the peer in the dynamic crypto map entry by hostname, as defined by the **name** command.
ip_address	Identifies the peer in the dynamic crypto map entry by IP address, as defined by the **name** command.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was changed from **crypto dynamic-map**.

Examples

The following example shows setting a peer for a dynamic-map named mymap to the IP address 10.0.0.1:

```
hostname(config)# crypto dynamic-map mymap 10 set peer 10.0.0.1
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto dynamic-map	Clears all configuration for all the dynamic crypto maps.
show running-config crypto dynamic-map	Displays all configuration for all the dynamic crypto maps.

crypto dynamic-map set pfs

To define a dynamic crypto map entry, use the **crypto dynamic-map set pfs** command in global configuration mode. T o remove the access list from a crypto map entry, use the **no** form of this command. See the **crypto map set pfs** command for additional information about this command.

crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set pfs [group1 | group2 | group5 | group 7]**

no crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set pfs [group1 | group2 | group5 | group 7]**

Syntax Description

dynamic-map-name	Specifies the name of the dynamic crypto map set.
dynamic-seq-num	Specifies the sequence number that corresponds to the dynamic crypto map entry.
group1	Specifies that IPSec should use the 768-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group2	Specifies that IPSec should use the 1024-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group5	Specifies that IPSec should use the 1536-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group7	Specifies that IPSec should use group7 (ECC) where the elliptical curve field size is 163-bits, for example, with the MovianVPN client.
set pfs	Configures IPSec to ask for perfect forward secrecy when requesting new security associations for this dynamic crypto map entry or configures IPSec to require PFS when receiving requests for new security associations.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was changed from **crypto dynamic-map**.

Usage Guidelines

The **crypto dynamic-map** commands, such as **match address**, **set peer**, and **set pfs** are described with the **crypto map** commands. If the peer initiates the negotiation and the local configuration specifies PFS, the peer must perform a PFS exchange or the negotiation fails. If the local configuration does not specify a group, the FWSM assumes a default of group2. If the local configuration does not specify PFS, it accepts any offer of PFS from the peer.

When interacting with the Cisco VPN client, the FWSM does not use the PFS value, but instead uses the value negotiated during Phase 1.

Examples

The following example specifies that PFS should be used whenever a new security association is negotiated for the crypto dynamic-map mymap 10. The group specified is group 2:

```
hostname(config)# crypto dynamic-map mymap 10 set pfs group2
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto dynamic-map	Clears all configuration for all the dynamic crypto maps.
show running-config crypto dynamic-map	Displays all configuration for all the dynamic crypto maps.

crypto dynamic-map set reverse route

To define a dynamic crypto map entry, use the **crypto dynamic-map set reverse route** command in global configuration mode. To remove the access list from a crypto map entry, use the **no** form of this command. See the **crypto map set reverse-route** command for additional information about this command.

crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set reverse route**

no crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set reverse route**

Syntax Description

dynamic-map-name	Specifies the name of the crypto map set.
dynamic-seq-num	Specifies the number you assign to the crypto map entry.

Defaults

The default value for this command is off.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was changed from **crypto dynamic-map**.

Examples

The following command enables RRI for the crypto dynamic-map named mymap:

```
hostname(config)# crypto dynamic-map mymap 10 set reverse route
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto dynamic-map	Clears all configuration for all the dynamic crypto maps.
show running-config crypto dynamic-map	Displays all configuration for all the dynamic crypto maps.

crypto dynamic-map set security-association lifetime

To define a dynamic crypto map entry, use the **crypto dynamic-map set security-association lifetime** command in global configuration mode. To remove the access list from a crypto map entry, use the **no** form of this command . See the **crypto map set security-association lifetime** command for additional information about this command.

crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set security-association lifetime seconds** *seconds* | **kilobytes** *kilobytes*

no crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set security-association lifetime seconds** *seconds* | **kilobytes** *kilobytes*

Syntax Description

dynamic-map-name	Specifies the name of the dynamic crypto map set.
dynamic-seq-num	Specifies the sequence number that corresponds to the dynamic crypto map entry.
kilobytes	Specifies the volume of traffic (in kilobytes) that can pass between peers using a given security association before that security association expires. The default is 4,608,000 kilobytes.
seconds	Specifies the number of seconds a security association will live before it expires. The default is 28,800 seconds (eight hours).

Defaults

The default number of kilobytes is 4,608,000; the default number of seconds is 28,800.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was changed from **crypto dynamic-map**.

Examples

The following command specifies a security association lifetime in seconds for crypto dynamic-map mymap:

```
hostname(config)# crypto dynamic-map mymap 10 set security-association lifetime seconds
1400
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto dynamic-map	Clears all configuration for all the dynamic crypto maps.
show running-config crypto dynamic-map	Displays all configuration for all the dynamic crypto maps.

crypto dynamic-map set transform-set

To define a dynamic crypto map entry, use the **crypto dynamic-map set transform-set** command in global configuration mode. To remove the access list from a crypto map entry, use the **no** form of this command. See the **crypto map set transform-set** command for additional information about this command.

crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set transform-set** *transform-set-name1* [*… transform-set-name9*]

no crypto dynamic-map *dynamic-map-name dynamic-seq-num* **set transform-set** *transform-set-name1* [*… transform-set-name9*]

Syntax Description

dynamic-map-name	Specifies the name of the dynamic crypto map set.
dynamic-seq-num	Specifies the sequence number that corresponds to the dynamic crypto map entry.
transform-set-name1 *transform-set-name9*	Identifies the transform set to be used with the dynamic crypto map entry (the names of transform sets defined using the **crypto ipsec** command).

Note The **crypto map set transform-set** command is required for dynamic crypto map entries. All you need in the entry is a transform set.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was changed from **crypto dynamic-map**.

Examples

The following command specifies two transform sets (tfset1 and tfset2) for the crypto dynamic-map mymap:

```
hostname(config)# crypto dynamic-map mymap 10 set transform-set tfset1 tfset2
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto dynamic-map	Clears all configuration for all the dynamic crypto maps.
show running-config crypto dynamic-map	Displays all configuration for all the dynamic crypto maps.

crypto ipsec df-bit

To configure DF-bit policy for IPSec packets, use the **crypto ipsec df-bit** command in global configuration mode.

crypto ipsec df-bit [**clear-df** | **copy-df** | **set-df**] *interface*

Syntax Description

clear-df	(Optional) Specifies that the outer IP header will have the DF bit cleared and that the FWSM may fragment the packet to add the IPSec encapsulation.
copy-df	(Optional) Specifies that the FWSM will look in the original packet for the outer DF bit setting.
set-df	(Optional) Specifies that the outer IP header will have the DF bit set; however, the FWSM may fragment the packet if the original packet had the DF bit cleared.
interface	Specifies an interface name.
token	Indicates a token-based server for user authentication is used.

Defaults

This command is disabled by default. If this command is enabled without a specified setting, the FWSM uses the **copy-df** setting as default.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The DF bit with IPSec tunnels feature lets you specify whether the FWSM can clear, set, or copy the Don't Fragment (DF) bit from the encapsulated header. The DF bit within the IP header determines whether a device is allowed to fragment a packet.

Use the **crypto ipsec df-bit** command in global configuration mode to configure the FWSM to specify the DF bit in an encapsulated header.

When encapsulating tunnel mode IPSec traffic, use the **clear-df** setting for the DF bit. This setting lets the device send packets larger than the available MTU size. Also this setting is appropriate if you do not know the available MTU size.

Examples

The following example, entered in global configuration mode, specifies sets the IPSec DF policy to **clear-df**:

```
hostname(config)# crypto ipsec df-bit clear-df inside
hostname(config)#
```

Related Commands

Command	Description
crypto ipsec fragmentation	Configures the fragmentation policy for IPSec packets.
show crypto ipsec df-bit	Displays the DF-bit policy for a specified interface.
show crypto ipsec fragmentation	Displays the fragmentation policy for a specified interface.

crypto ipsec fragmentation

To configure the fragmentation policy for IPSec packets, use the **crypto ipsec fragmentation** command in global configuration mode.

crypto ipsec fragmentation {**after-encryption** | **before-encryption**} *interface*

Syntax Description

after-encryption	Specifies the FWSM to fragment IPSec packets that are close to the maximum MTU size after encryption (disables pre-fragmentation).
before-encryption	Specifies the FWSM to fragment IPSec packets that are close to the maximum MTU size before encryption (enables pre-fragmentation).
interface	Specifies an interface name.
token	Indicate a token-based server for user authentication is used.

Defaults

This feature is enabled by default.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

When a packet is near the size of the MTU of the outbound link of the encrypting FWSM, and it is encapsulated with IPSec headers, it is likely to exceed the MTU of the outbound link. This causes packet fragmentation after encryption, which makes the decrypting device reassemble in the process path. Pre-fragmentation for IPSec VPNs increases the decrypting device performance by letting it operate in the high performance CEF path instead of the process path.

Pre-fragmentation for IPSec VPNs lets an encrypting device predetermine the encapsulated packet size from information available in transform sets, which are configured as part of the IPSec SA. If the device predetermines that the packet will exceed the MTU of the output interface, the device fragments the packet before encrypting it. This avoids process level reassembly before decryption and helps improve decryption performance and overall IPsec traffic throughput.

Examples

The following example, entered in global configuration mode, enables pre-fragmentation for IPSec packets on the interface:

```
hostname(config)# crypto ipsec fragmentation before-encryption mgmt
hostname(config)#
```

The following example, entered in global configuration mode, disables pre-fragmentation for IPSec packets on the interface:

```
hostname(config)# crypto ipsec fragmentation after-encryption mgmt
hostname(config)#
```

Related Commands

Command	Description
crypto ipsec df-bit	Configures the DF-bit policy for IPSec packets.
show crypto ipsec fragmentation	Displays the fragmentation policy for IPSec packets.
show crypto ipsec df-bit	Displays the DF-bit policy for a specified interface.

crypto ipsec security-association lifetime

To configure global lifetime values, use the **crypto ipsec security-association lifetime** command in global configuration mode. To reset a crypto IPSec entry lifetime value to the default value, use the **no** form of this command.

crypto ipsec security-association lifetime {**seconds** *seconds* | **kilobytes** *kilobytes*}

no crypto ipsec security-association lifetime {**seconds** *seconds* | **kilobytes** *kilobytes*}

Syntax Description

kilobytes	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 kilobytes.The default is 4,608,000 kilobytes.
seconds	Specifies the number of seconds a security association will live before it expires. The range is 120 to 214783647 seconds. The default is 28,800 seconds (eight hours).
token	Indicate a token-based server for user authentication is used.

Defaults

The default number of kilobytes is 4,608,000; the default number of seconds is 28,800.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

The **crypto ipsec security-association lifetime** command changes global lifetime values used when negotiating IPSec security associations.

IPSec security associations use shared secret keys. These keys and their security associations time out together.

Assuming that the particular crypto map entry has no lifetime values configured, when the FWSM requests new security associations during negotiation, it specifies its global lifetime value in the request to the peer; it uses this value as the lifetime of the new security associations. When the FWSM receives a negotiation request from the peer, it uses the smaller of the lifetime value proposed by the peer or the locally configured lifetime value as the lifetime of the new security associations.

There are two lifetimes: a “timed” lifetime and a “traffic-volume” lifetime. The security association expires after the first of these lifetimes is reached.

The FWSM lets you change crypto map, dynamic map, and ipsec settings on the fly. If you do so, the FWSM 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.

To change the global timed lifetime, use the **crypto ipsec security-association lifetime seconds** command. The timed lifetime causes the security association to time out after the specified number of seconds have passed.

To change the global traffic-volume lifetime, use the **crypto ipsec security-association lifetime kilobytes** command. The traffic-volume lifetime causes the security association to time out after the specified amount of traffic (in kilobytes) has been protected by the security associations key.

Shorter lifetimes can make it harder to mount a successful key recovery attack, because the attacker has less data encrypted under the same key to work with. However, shorter lifetimes require more CPU processing time for establishing new security associations.

The security association (and corresponding keys) expires according to whichever occurs sooner, either after the number of seconds has passed or after the amount of traffic in kilobytes has passed.

Examples

The following example specifies a global timed lifetime for security associations:

```
hostname(config)# crypto ipsec-security association lifetime seconds 240
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all IPSec configuration, such as global lifetimes and transform sets.
show running-config crypto map	Displays all configuration for all the crypto maps.

crypto ipsec transform-set

To define a transform set, use the **crypto ipsec transform-set** command in global configuration mode. With this command, you identify the IPSec encryption and hash algorithms to be used by the transform set. To remove a transform set, use the **no** form of this command.

crypto ipsec *map-name seq-num* **transform-set** *transform-set-name transform1 [transform2]*

no crypto ipsec *map-name seq-num* **transform-set** *transform-set-name*

Syntax Description

esp-aes	Specifying this option means that IPSec messages protected by this transform are encrypted using AES with a 128-bit key.
esp-aes-192	Specifying this option means that IPSec messages protected by this transform are encrypted using AES with a 192-bit key.
esp-aes-256	Specifying this option means that IPSec messages protected by this transform are encrypted using AES with a 256-bit key.
esp-des	Specifying this option means that IPSec messages protected by this transform with encryption using 56-bit DES-CBC.
esp-3des	Specifying this option means that IPSec messages protected by this transform are encrypted using the Triple DES algorithm.
esp-none	Specifying this option means that IPSec messages do not use HMAC authentication.
esp-null	Specifying this option means that IPSec messages are not encrypted using the IPSec security protocol (ESP) only.
esp-md5-hmac	Specifying this option means that IPSec messages protected by this transform are using MD5/HMAC-128 as the hash algorithm.
esp-sha-hmac	Specifying this option means that IPSec messages protected by this transform are using SHA/HMAC-160 as the hash algorithm.
map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.
transform1, transform2	Specifies up to two transforms. Transforms define the IPSec security protocol(s) and algorithm(s). Each transform represents an IPSec security protocol (ESP), plus the algorithm to use, either [**esp-aes | esp-aes-192 | esp-aes-256 | esp-des | esp-3des | esp-null**] or [**esp-md5-hmac | esp-sha-hmac**] as defined in this syntax table.
transform-set-name	Specifies the name of the transform set to create or modify.
token	Indicate a token-based server for user authentication is used.

Defaults

The default encryption algorithm is esp-3des (Triple DES).

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

A transform set specifies one or two IPSec security protocols and specifies which algorithms to use with the selected security protocol. During the IPSec security association negotiation, the peers agree to use a particular transform set when protecting a particular data flow.

IPSec messages can be protected by a transform set using AES with a 128-bit key, 192-bit key, or 256-bit key.

Due to the large key sizes provided by AES, ISAKMP negotiation should use Diffie-Hellman group 5 instead of group 1 or group 2. To do this, use the **isakmp policy priority group 5** command.

You can configure multiple transform sets, and then specify one or more of these transform sets in a crypto map entry. The transform set defined in the crypto map entry in the IPSec security association negotiation protects the data flows specified by the access list of that crypto map entry. During the negotiation, the peers search for a transform set that is the same at both peers. When the FWSM finds such a transform set, it applies it to the protected traffic as part of the IPSec security associations of both peers.

Each transform-set represents an algorithm to use for encryption or authentication. When the particular transform set is used during negotiations for IPSec security associations, the entire transform set (the combination of protocols, algorithms, and other settings) must match a transform set at the remote peer.

In a transform set, you can specify just an ESP encryption transform or both an ESP encryption transform and an ESP authentication transform.

Examples of acceptable transform combinations are as follows:

- **esp-des**
- **esp-des** and **esp-md5-hmac**

If one or more transforms are specified in the **crypto ipsec transform-set** command for an existing transform set, the specified transforms replace the existing transforms for that transform set.

Examples

The following example configures two transform sets: one named t1, using DES for encryption and SHA/HMAC-160 as the hash algorithm, and the other named standard, using AES 192 for encryption and MD5/HMAC-128 as the hash algorithm:

```
hostname(config)# crypto ipsec transform-set t1 esp-des esp-sha-hmac
hostname(config)# crypto ipsec transform-set standard esp-aes-192 esp-md5-hmac
hostname(config)
```

Related Commands

Command	Description
clear configure crypto	Clears all IPSec configuration (that is, global lifetimes and transform sets.
show running-config crypto map	Displays all configuration for all the crypto maps.

crypto key generate dsa

To generate DSA key pairs for identity certificates, use the **crypto key generate dsa** command in global configuration mode.

crypto key generate dsa {**label** *key-pair-label*} [**modulus** *size*] [**noconfirm**]

Syntax Description

label *key-pair-label*	Specifies the name to be associated with the key pair(s); maximum label length is 128 characters. DSA requires a label.
modulus *size*	(Optional) Specifies the modulus size of the key pair(s): 512, 768, 1024. The default modulus size is 1024.
noconfirm	(Optional) Suppresses all interactive prompting.

Defaults

The default modulus size is 1024.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Use the **crypto key generate dsa** command to generate DSA key pairs to support SSL, SSH, and IPSec connections. The generated key pairs are identified by labels that you provide as part of the command syntax. If you do not provide a label, the FWSM displays an error message.

Examples

The following example, entered in global configuration mode, generates an DSA key pair with the label mypubkey:

```
hostname(config)# crypto key generate dsa label mypubkey
INFO: The name for the keys will be: mypubkey
hostname(config)#
```

The following example, entered in global configuration mode, inadvertently attempts to generate a duplicate DSA key pair with the label mypubkey:

```
hostname(config)# crypto key generate dsa label mypubkey
WARNING: You already have dSA keys defined named mypubkey
Do you really want to replace them? [yes/no] no
ERROR: Failed to create new DSA keys named mypubkey
hostname(config)#
```

Related Commands

Command	Description
crypto key zeroize	Removes the DSA key pairs.
show crypto key mypubkey	Displays the DSA key pairs.

crypto key generate rsa

To generate RSA key pairs for identity certificates, use the **crypto key generate rsa** command in global configuration mode.

crypto key generate rsa [**usage-keys | general-keys**] [**label** *key-pair-label*] [**modulus** *size*] [**noconfirm**]

Syntax Description

general-keys	(Optional) Generates a single pair of general purpose keys. This is the default key-pair type.
label *key-pair-label*	(Optional) Specifies the name to be associated with the key pair(s). This key pair must be uniquely labeled. If you attempt to create another key pair with the same label, the FWSM displays an warning message. If no label is provided when the key is generated, the key pair is statically named <Default-RSA-Key>.
modulus *size*	(Optional) Specifies the modulus size of the key pair(s): 512, 768, 1024, and 2048. The default modulus size is 1024.
noconfirm	(Optional) Suppresses all interactive prompting.
usage-keys	(Optional) Generates two key pairs, one for signature use and one for encryption use. This implies that two certificates for the corresponding identity are required.

Defaults

The default key-pair type is **general key**. The default modulus size is 1024.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

Use the **crypto key generate rsa** command to generate RSA key pairs to support SSL, SSH, and IPSec connections. The generated key pairs are identified by labels that you can provide as part of the command syntax. Trustpoints that do not reference a key pair can use the default one <Default-RSA-Key>. SSH connections always use this key. This does not affect SSL, since SSL generates its own cert/key dynamically, unless a trustpoint has one configured.

Examples

The following example, entered in global configuration mode, generates an RSA key pair with the label mypubkey:

```
hostname(config)# crypto key generate rsa label mypubkey
INFO: The name for the keys will be: mypubkey
Keypair generation process
hostname(config)#
```

The following example, entered in global configuration mode, inadvertently attempts to generate a duplicate RSA key pair with the label mypubkey:

```
hostname(config)# crypto key generate rsa label mypubkey
WARNING: You already have RSA keys defined named mypubkey
Do you really want to replace them? [yes/no] no
ERROR: Failed to create new RSA keys named mypubkey
hostname(config)#
```

The following example, entered in global configuration mode, generates an RSA key pair with the default label:

```
hostname(config)# crypto key generate rsa
INFO: The name for the keys will be: <Default-RSA-Key>
Keypair generation process begin. Please wait...
hostname(config)#
```

Related Commands

Command	Description
crypto key zeroize	Removes RSA key pairs.
show crypto key mypubkey	Displays the RSA key pairs.

crypto key zeroize

To remove the key pairs of the indicated type (rsa or dsa), use the **crypto key zeroize** command in global configuration mode.

crypto key zeroize {**rsa** | **dsa**} [**label** *key-pair-label*] [**default**] [**noconfirm**]

Syntax Description

default	(Optional) Removes RSA key pairs with no labels. This keyword is legal only with RSA key pairs.
dsa	Specifies DSA as the key type.
label *key-pair-label*	(Optional) Removes the key pairs of the indicated type (rsa or dsa). If you do not provide a label, the FWSM removes all key pairs of the indicated type.
noconfirm	(Optional) Suppresses all interactive prompting.
rsa	Specifies RSA as the key type.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Examples

The following example, entered in global configuration mode, removes all RSA key pairs:

```
hostname(config)# crypto key zeroize rsa
WARNING: All RSA keys will be removed.
WARNING: All router certs issued using these keys will also be removed.

Do you really want to remove these keys? [yes/no] y
hostname(config)#
```

Related Commands

Command	Description
crypto key generate dsa	Generates DSA key pairs for identity certificates.
crypto key generate rsa	Generate RSA key pairs for identity certificates.

crypto map interface

Use the **crypto map interface** command in global configuration mode to apply a previously defined crypto map set to an interface. To remove the crypto map set from the interface, use the **no** form of this command.

crypto map *map-name* **interface** *interface-name*

no crypto map *map-name* **interface** *interface-name*

Syntax Description

interface-name	Specifies the interface for the FWSM to use for establishing tunnels with VPN peers. If ISAKMP is enabled, and you are using a certificate authority to obtain certificates, this should be the interface with the address specified in the CA certificates.
map-name	Specifies the name of the crypto map set.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

Use this command to assign a crypto map set to any active FWSM interface. The FWSM supports IPSec termination on any and all active interfaces. You must assign a crypto map set to an interface before that interface can provide IPSec services.

You can assign only one crypto map set to an interface. If multiple crypto map entries have the same *map-name* but a different *seq-num*, they are part of the same set and are all applied to the interface. The FWSM evaluates the crypto map entry with the lowest *seq-num* first.

Note The FWSM lets you change crypto map, dynamic map, and IPsec settings on the fly. If you do so, the FWSM 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.

Every static crypto map must define three parts: an access list, a transform set, and an IPsec peer. If one of these is missing, the crypto map is incomplete and the FWSM moves on to the next entry. However, if the crypto map matches on the access list but not on either or both of the other two requirements, this FWSM drops the traffic.

Use the **show running-config crypto map** command to ensure that every crypto map is complete. To fix an incomplete crypto map, remove the crypto map, add the missing entries, and reapply it.

Examples

The following example, entered in global configuration mode, assigns the crypto map set named mymap to the outside interface. When traffic passes through the outside interface, the FWSM evaluates it against all the crypto map entries in the mymap set. When outbound traffic matches an access list in one of the mymap crypto map entries, the FWSM forms a security association using the configuration of that crypto map entry.

```
hostname(config)# crypto map mymap interface outside
```

The following example shows the minimum required crypto map configuration:

```
hostname(config)# crypto map mymap 10 ipsec-isakmp
hostname(config)# crypto map mymap 10 match address 101
hostname(config)# crypto map mymap set transform-set my_t_set1
hostname(config)# crypto map mymap set peer 10.0.0.1
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map ipsec-isakmp dynamic

To require a given crypto map entry to refer to a pre-existing dynamic crypto map, use the **crypto map ipsec-isakmp dynamic** command in global configuration mode. To remove the cross reference, use the **no** form of this command .

[no] crypto map *map-name seq-num* **ipsec-isakmp dynamic** *dynamic-map-name*

Syntax Description

dynamic-map-name	Specifies the name of the crypto map entry that refers to a pre-existing dynamic crypto map.
ipsec-isakmp	Indicates that IKE establishes the IPSec security associations for this crypto map entry.
map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was modified to remove the **ipsec-manual** keyword.

Usage Guidelines

Use the **crypto dynamic-map** command to create dynamic crypto map entries. After you create a dynamic crypto map set, use the **crypto map ipsec-isakmp dynamic** command to add the dynamic crypto map set to a static crypto map.

After you define crypto map entries, you can use the **crypto map interface** command to assign the dynamic crypto map set to interfaces.

Dynamic crypto maps provide two functions: filtering/classifying traffic to protect, and defining the policy to apply to that traffic. The first use affects the flow of traffic on an interface; the second affects the negotiation performed (via IKE) on behalf of that traffic.

IPSec dynamic crypto maps identify the following:

- The traffic to protect
- IPSec peer(s) with which to establish a security association
- Transform sets to use with the protected traffic

- How to use or manage keys and security associations

A crypto map set is a collection of crypto map entries, each with a different sequence number (seq-num) but the same map name. Therefore, for a given interface, you could have certain traffic forwarded to one peer with specified security applied to that traffic, and other traffic forwarded to the same or a different peer with different IPSec security applied. To accomplish this you create two crypto map entries, each with the same map name, but each with a different sequence number.

The number you assign as the seq-num argument should not be arbitrary. This number ranks multiple crypto map entries within a crypto map set. A crypto map entry with a lower seq-num is evaluated before a map entry with a higher seq-num; that is, the map entry with the lower number has a higher priority.

Note When you link the crypto map to a dynamic crypto map, you must specify the dynamic crypto map. This links the crypto map to an existing dynamic crypto map that was previously defined using the **crypto dynamic-map** command. Now any changes you make to the crypto map entry after it has been converted, will not take affect. For example, a change to the set peer setting does not take effect. However, the FWSM stores the change while it is up. When the dynamic crypto map is converted back to the crypto map, the change is effective and appears in the output of the **show running-config crypto map** command. The FWSM maintains these settings until it reboots.

Examples

The following command, entered in global configuration mode, configures the crypto map mymap to refer to a dynamic crypto map named test:

```
hostname(config)# crypto map mymap ipsec-isakmp dynamic test
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map match address

To assign an access list to a crypto map entry, use the **crypto map match address** command in global configuration mode.To remove the access list from a crypto map entry, use the **no** form of this command.

crypto map *map-name seq-num* **match address** *acl_name*

no crypto map *map-name seq-num* **match address** *acl_name*

Syntax Description

acl_name	Specifies the name of the encryption access list. This name should match the name argument of the named encryption access list being matched.
map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

This command is required for all static crypto map entries. If you are defining a dynamic crypto map entry (with the **crypto dynamic-map** command), this command is not required but is strongly recommended. You would use the **access-list** command to define this access list.

IPSec uses this access list to differentiate the traffic to protect by IPSec crypto from the traffic that does not need protection. (Traffic permitted by the access list is protected. Traffic denied by the access list is not protected in the context of the corresponding crypto map entry.)

Note The crypto access list does not determine whether to permit or deny traffic through the interface. An access list applied directly to the interface with the **access-group** command makes that determination.

In transparent mode, the destination address should be the IP address of the FWSM, the management address. Only tunnels to the FWSM are allowed in transparent mode.

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map set connection-type

To specify the connection type for the Backup Site-to-Site feature for this crypto map entry, use the **crypto map set connection-type** command in global configuration mode. To return to the default setting, use the **no** form of this command.

crypto map *map-name seq-num* **set connection-type {answer-only | originate-only | bidirectional}**

no crypto map *map-name seq-num* **set connection-type {answer-only | originate-only | bidirectional}**

Syntax Description

answer-only	Indicates that this peer can only respond to inbound IKE connections for Site-to-Site connections based on this crypto map entry. It cannot originate connection requests. This keyword is the only available option for transparent firewall mode.
bidirectional	Indicates that this peer can accept and originate connections based on this crypto map entry. This is the default connection type for all Site-to-Site connections. This keyword is not available in transparent firewall mode.
map-*name*	Specifies the name of the crypto map set.
originate-only	Indicates that this peer can only originate connections based on this crypto map entry. It cannot accept inbound connections. This keyword is not available in transparent firewall mode.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

The default setting is bidirectional.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Examples

The following example, entered in global configuration mode, configures the crypto map mymap and sets the connection-type to bidirectional:

```
hostname(config)# crypto map mymap 10 set connection-type bidirectional
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map set peer

To specify an IPSec peer in a crypto map entry, use the **crypto map set peer** command in global configuration mode. To remove an IPSec peer from a crypto map entry, use the **no** form of this command.

crypto map *map-name seq-num* **set peer** {*ip_address* | *hostname*}{*...ip_address* | *hostname10*}

no crypto map *map-name seq-num* **set peer** {*ip_address* | *hostname*}{*...ip_address* | *hostname10*}

Syntax Description

hostname	Specifies a peer by its host name as defined by the FWSM **name** command.
ip_address	Specifies a peer by its IP address.
map-name	Specifies the name of the crypto map set.
peer	Specifies an IPSec peer in a crypto map entry either by hostname of IP address.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was modified to allow up to 10 peer addresses.

Usage Guidelines

This command is required for all static crypto maps. If you are defining a dynamic crypto map (with the **crypto dynamic-map** command), this command is not required, and in most cases is not used because, in general, the peer is unknown.

For LAN-to-LAN connections, you can use multiple peers only with originator-only connection type. Configuring multiple peers is equivalent to providing a fallback list. For each tunnel, the FWSM attempts to negotiate with the first peer in the list. If that peer does not respond, the FWSM works its way down the list until either a peer responds or there are no more peers in the list. You can set up multiple peers only when using the backup LAN-to-LAN feature (that is, when the crypto map is originate-only type).

Examples

The following example, entered in global configuration mode, shows a crypto map configuration using IKE to establish the security associations. In this example, you can set up a security association to either the peer at 10.0.0.1 or the peer at 10.0.0.2:

```
hostname(config)# crypto map mymap 10 ipsec-isakmp
hostname(config)# crypto map mymap 10 match address 101
hostname(config)# crypto map mymap 10 set transform-set my_t_set1
hostname(config)# crypto map mymap 10 set peer 10.0.0.1 10.0.0.2
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map set pfs

To set IPSec to ask for perfect forward secrecy (PFS) when requesting new security associations for this crypto map entry, or to set that IPSec requires PFS when receiving requests for new security associations, use the **crypto map set pfs** command in global configuration mode. To specify that IPSec should not request PFS, use the **no** form of this command.

crypto map *map-name seq-num* **set pfs [group1 | group2 | group5 | group7]**

no crypto map *map-name seq-num* **set pfs [group1 | group2 | group5 | group7]**

Syntax Description

group1	Specifies that IPSec should use the 768-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group2	Specifies that IPSec should use the 1024-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group5	Specifies that IPSec should use the 1536-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group7	Specifies that IPSec should use group7 (ECC) where the elliptical curve field size is 163-bits, for example, with the MovianVPN client.
map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

By default PFS is not set.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.
3.1(1)	This command was modified to add Diffie-Hellman group 7.

Usage Guidelines

With PFS, every time a new security association is negotiated, a new Diffie-Hellman exchange occurs, which requires additional processing time. PFS adds another level of security because if one key is ever cracked by an attacker, only the data sent with that key is compromised.

During negotiation, this command causes IPSec to request PFS when requesting new security associations for the crypto map entry. If the **set pfs** statement does not specify a group, the FWSM sends the default (group2).

If the peer initiates the negotiation and the local configuration specifies PFS, the peer must perform a PFS exchange or the negotiation fails. If the local configuration does not specify a group, the FWSM assumes a default of group2. If the local configuration specifies group2, group5, or group7, that group must be part of the offer from the peer or the negotiation fails.

For a negotiation to succeed PFS has to be set on both ends. If set, the groups have to be an exact match; The FWSM does not accept just any offer of PFS from the peer.

The 1536-bit Diffie-Hellman prime modulus group, group5, provides more security than group1, or group2, but requires more processing time than the other groups.

Diffie-Hellman Group 7 generates IPSec SA keys, where the elliptical curve field size is 163 bits. You can use this option with any encryption algorithm. This option is intended for use with the MovianVPN client, but you can use it with any peers that support Group 7 (ECC).

When interacting with the Cisco VPN client, the FWSM does not use the PFS value, but instead uses the value negotiated during Phase 1.

Examples

The following example, entered in global configuration mode, specifies that PFS should be used whenever a new security association is negotiated for the crypto map “mymap 10”:

```
hostname(config)# crypto map mymap 10 ipsec-isakmp
hostname(config)# crypto map mymap 10 set pfs group2
```

Related Commands

Command	Description
clear isakmp sa	Deletes the active IKE security associations.
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.
tunnel-group	Configures tunnel-groups and their parameters.

crypto map set phase1 mode

To specify the IKE mode for phase 1 when initiating a connection to either main or aggressive, use the **crypto map set phase1mode** command in global configuration mode. To remove the setting for phase 1 IKE negotiations, use the **no** form of this command. Including a Diffie-Hellman group with aggressive mode is optional. If one is not included, the FWSM uses group 2.

crypto map *map-name seq-num* **set phase1mode {main | aggressive [group1 | group2 | group5 | group7]}**

no crypto map *map-name seq-num* **set phase1mode {main | aggressive [group1 | group2 | group5 | group7]}**

Syntax Description

aggressive	Specifies aggressive mode for phase one IKE negotiations
group1	Specifies that IPSec should use the 768-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group2	Specifies that IPSec should use the 1024-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group5	Specifies that IPSec should use the 1536-bit Diffie-Hellman prime modulus group when performing the new Diffie-Hellman exchange.
group7	Specifies that IPSec should use group7 (ECC) where the elliptical curve field size is 163-bits, for example, with the MovianVPN client.
main	Specifies main mode for phase one IKE negotiations.
map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

Default phase one mode is **main**.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

This command works only in initiator mode; not in responder mode.

Examples

The following example, entered in global configuration mode, configures the crypto map mymap and sets the phase one mode to aggressive, using group 2:

```
hostname(config)# crypto map mymap 10 set phase1mode aggressive group2
hostname(config)#
```

Related Commands

Command	Description
clear isakmp sa	Delete the active IKE security associations.
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map set reverse-route

To enable RRI for any connection based on this crypto map entry, use the **crypto map set reverse-route** command in global configuration mode. To disable reverse route injection for any connection based this crypto map entry, use the **no** form of this command.

crypto map *map-name seq-num* **set reverse-route**

no crypto map *map-name seq-num s***et reverse-route**

Syntax Description

map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

The default setting for this command is off.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

The FWSM can automatically add static routes to the routing table and announce these routes to its private network or border routers using OSPF.

Examples

The following example, entered in global configuration mode, enables RRI for the crypto map named mymap:

```
hostname(config)# crypto map mymap 10 set reverse-route
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map set security-association lifetime

To override (for a particular crypto map entry) the global lifetime value, which is used when negotiating IPSec security associations, use the **crypto map set security-association lifetime** command in global configuration mode. To reset the lifetime value of a crypto map entry to the global value, use the **no** form of this command.

crypto map *map-name seq-num* **set security-association lifetime {seconds** *seconds* | **kilobytes** *kilobytes*}

no crypto map *map-name seq-num* **set security-association lifetime** {**seconds** *seconds* | **kilobytes** *kilobytes*}

Syntax Description

kilobytes	Specifies the volume of traffic (in kilobytes) that can pass between peers using a given security association before that security association expires. The default is 4,608,000 kilobytes.
map-name	Specifies the name of the crypto map set.
seconds	Specifies the number of seconds a security association will live before it expires. The default is 28,800 seconds (eight hours).
seq-num	Specifies the number you assign to the crypto map entry.

Defaults

The default number of kilobytes is 4,608,000; the default number of seconds is 28,800.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

The security associations of a crypto map 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 FWSM 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 FWSM 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.

There are two lifetimes: a “timed” lifetime and a “traffic-volume” lifetime. The session keys/security association expires after the first of these lifetimes is reached. You can specify both with one command.

Note The FWSM lets you change crypto map, dynamic map, and ipsec settings on the fly. If you do so, the FWSM 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.

To change the timed lifetime, use the **crypto map set security-association lifetime seconds** command. The timed lifetime causes the keys and security association to time out after the specified number of seconds have passed.

Examples

The following command, entered in global configuration mode, specifies a security association lifetime in seconds and kilobytes for crypto map mymap:

```
hostname(config)# crypto map mymap 10 set security-association lifetime seconds 1400
kilobytes 3000000
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.

crypto map set transform-set

To specify the transform sets to use with the crypto map entry, use the **crypto map set transform-set** command in global configuration mode. To remove the specified transform sets from a crypto map entry, use the **no** form of this command.

crypto map *map-name seq-num* **set transform-set** *transform-set-name1* [*… transform-set-name9*]

no crypto map *map-name seq-num* **set transform-set** *transform-set-name1* [*… transform-set-name9*]

Syntax Description

map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.
transform-set-name1 *transform-set-name9*	Specifies the name(s) of the transform set(s), defined using the **crypto ipsec transform-set** command, to use for the crypto map. For an ipsec-isakmp or dynamic crypto map entry, you can specify up to nine transform sets.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

This command is required for all crypto map entries.

If the local FWSM initiates the negotiation, the transform sets are presented to the peer in the order specified in the **crypto map** command statement. If the peer initiates the negotiation, the local FWSM accepts the first transform set that matches one of the transform sets specified in the crypto map entry.

The first matching transform set that is found at both peers is used for the security association. If no match is found, IPSec does not establish a security association. The traffic is dropped because there is no security association to protect the traffic.

If you want to change the list of transform sets, respecify the new list of transform sets to replace the old list. This change is applied only to **crypto map** command statements that reference this transform set.

Any transform sets included in a **crypto map** command statement must previously have been defined using the **crypto ipsec transform-set** command.

Examples

The following example, entered in global configuration mode, specifies two transform sets (tfset1 and tfset2) for the crypto map mymap:

```
hostname(config)# crypto map mymap 10 set transform-set tfset1 tfset2
hostname(config)#
```

The following example, entered in global configuration mode, shows the minimum required crypto map configuration when the FWSM uses IKE to establish the security associations:

```
hostname(config)# crypto map mymap 10 ipsec-isakmp
hostname(config)# crypto map mymap 10 match address 101
hostname(config)# crypto map mymap set transform-set my_t_set1
hostname(config)# crypto map mymap set peer 10.0.0.1
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
crypto ipsec transform-set	Configures a transform-set.
show running-config crypto map	Displays the crypto map configuration.

crypto map set trustpoint

To specify the trustpoint that identifies the certificate to send for authentication during Phase 1 negotiations for the crypto map entry, use the **crypto map set trustpoint** command in global configuration mode. To remove a trustpoint from a crypto map entry, use the **no** form of this command.

crypto map *map-name seq-num* **set trustpoint** *trustpoint-name* **[chain]**

nocrypto map *map-name seq-num* **set trustpoint** *trustpoint-name* **[chain]**

Syntax Description

chain	(Optional) Sends a certificate chain. A CA certificate chain includes all CA certificates in a hierarchy of certificates from the root certificate to the identity certificate. The default value is disable (no chain).
map-name	Specifies the name of the crypto map set.
seq-num	Specifies the number you assign to the crypto map entry.
trustpoint-name	Identifies the certificate to be sent during Phase 1 negotiations. The default is none.

Defaults

The default value is none.

Command Modes

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

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

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

This crypto map command is valid only for initiating a connection. For information on the responder side, see the **tunnel-group** commands.

Examples

The following example, entered in global configuration mode, specifies a trustpoint named tpoint1 for crypto map mymap and includes the chain of certificates:

```
hostname(config)# crypto map mymap 10 set trustpoint tpoint1 chain
hostname(config)#
```

Related Commands

Command	Description
clear configure crypto map	Clears all configuration for all crypto maps.
show running-config crypto map	Displays the crypto map configuration.
tunnel-group	Configures tunnel groups.