Security Certifications Compliance

Security Certifications Compliance

United States federal government agencies are sometimes required to use only equipment and software complying with security standards established by the U.S. Department of Defense and global certification organizations. The Firepower 4100/9300 chassis supports compliance with several of these security certification standards.

See the following topics for steps to enable features that support compliance with these standards:

- Enable FIPS Mode
- Enable Common Criteria Mode
- Configure IPSec Secure Channel, on page 3
- Configure Static CRL for a Trustpoint, on page 9
- About the Certificate Revocation List Check, on page 10
- Configure CRL Periodic Download, on page 14
- Setting the Date and Time Using NTP
- Set the LDAP Key Ring Certificate, on page 15
- Configure the IP Access List
- Configure Minimum Password Length Check
- Set the Maximum Number of Login Attempts

Note Note that these topics discuss enabling certifications compliance on the Firepower 4100/9300 chassis only. Enabling certification compliance on the Firepower 4100/9300 chassis does not automatically propagate compliance to any of its attached logical devices.

Generate the SSH Host Key

Prior to FXOS release 2.0.1, the existing SSH host key created during initial setup of a device was hard-coded to 1024 bits. To comply with FIPS and Common Criteria certification, you must destroy this old host key and generate a new one. See Enable FIPS Mode or Enable Common Criteria Mode for more information.

Perform these steps to destroy the old SSH host key and generate a new certifications-compliant one.

Procedure

Step 1 From the FXOS CLI, enter services mode:

scope system

scope services

Step 2 Delete the SSH host key:

delete ssh-server host-key

Step 3 Commit the configuration:

commit-buffer

Step 4 Set the SSH host key size to 2048 bits:

set ssh-server host-key rsa 2048

Step 5 Commit the configuration:

commit-buffer

Step 6 Create a new SSH host key:

create ssh-server host-key

commit-buffer

Step 7 Confirm the new host key size:

show ssh-server host-key

Host Key Size: 2048

Configure IPSec Secure Channel

IPSec is a framework of open standards developed by the Internet Engineering Task Force (IETF). It creates secure, authenticated, and reliable communication over IP networks. The IPSec security service provides:

- Connectionless Integrity – Assurance the received traffic has not been modified.
- Data origin authentication – Assurance the traffic is sent by legitimate party.
- Confidentiality (encryption) – Assurance the user's traffic is not examined by non-authorized parties.
- Access control – Prevention of unauthorized use of a resource.

IPSec Secure Channel supports the following algorithms:

- Phase 1

```
aes128gcm16-prfsha384-prfsha512-prfsha256-prfsha1-ecp256-ecp384-ecp521-modp2048-modp3072-modp4096
aes128-aes192-aes256-sha256-sha384-sha1_160-sha1-sha512-prfsha384-prfsha512-prfsha256-prfsha1-ecp256-ecp384-ecp521
aes128-aes192-aes256-sha256-sha384-sha1_160-sha1-sha512-prfsha384-prfsha512-prfsha256-prfsha1-modp2048-modp3072-modp4096
```

- Phase 2
  - Only AES SHA based encryption algorithms are supported. (DES and MD5 are not supported)
  - Supported DH groups are 14,15,16,19,20, and 21.

Note IPSec connections can only be initiated from FXOS. FXOS does not accept incoming IPSec connection requests.

IPsec tunnels are sets of SAs that FXOS establishes between peers. The SAs specify the protocols and algorithms to apply to sensitive data and also specify the keying material that the peers use. IPsec SAs control the actual transmission of user traffic. SAs are unidirectional, but are generally established in pairs (inbound and outbound).

IPSec on Chassis Manager has two modes:

Transport Mode
IP Header, IPSec Header, TCP Header, Data

Tunnel Mode
New IP Header, IPSec Header, Original IP Header, TCP Header, Data

IPSec's operation can be broken down into five main steps:

1. Traffic Selection – Interesting traffic which matches IPSec policy starts the IKE process. For example, traffic can be selected using src/dst host IP or subnet. Alternatively, user also can trigger IKE process through admin command.
2. IKE Phase 1 – authenticate IPSec peers and to setup a secure channel to enable IKE exchanges
3. IKE phase 2 – negotiate SAs to set up the IPSec tunnel. SA stands for Security Association, it is a relationship between IPSec end-points that describe what security services are used to protect data traffic.

4. Data transfer – Data packets are encrypted and encapsulated in IPSec header using parameters and keys stored in the SA
5. IPSec tunnel termination – IPSec SAs terminate through deletion or by timing out.

You can configure IPSec on your Firepower 4100/9300 chassis to provide end-to-end data encryption and authentication service on data packets going through the public network. This option is one of a number offered for achieving Common Criteria certification compliance on your system. For more information, see Security Certifications Compliance, on page 1.

Note

- If you are using an IPSec secure channel in FIPS mode, the IPSec peer must support RFC 7427.
- If you elect to configure enforcement of matching cryptographic key strength between IKE and SA connections (set sa-strength-enforcement to yes in the below procedure):

If SA enforcement is enabled	then when IKE negotiated key size is less then ESP negotiated key size, the connection fails. then when IKE negotiated key size is large or equal than ESP negotiated key size, SA enforcement check passes and the connection is successful.
If SA enforcement is disabled	then SA enforcement check passes and the connection is successful.

Perform these steps to configure an IPSec secure channel.

Procedure

Step 1 From the FXOS CLI, enter security mode:

scope security

Step 2 Create the keyring:

enter keyring ssp

! create certreq subject-name *subject-name* **ip** *ip*

Step 3 Enter the associated certificate request information:

enter certreq

Step 4 Set the country:

set country *country*

Step 5 Set the DNS:

set dns *dns*

Step 6 Set the email:

set e-mail *email*

Step 7 Set the IP information:

set ip *ip-address*

set ipv6 *ipv6*

Step 8 Set the locality:

set locality *locality*

Step 9 Set the organization name:

set org-name *org-name*

Step 10 Set the organization unit name:

set org-unit-name *org-unit-name*

Step 11 Set the password:

! set password

Step 12 Set the state:

set state *state*

Step 13 Set the subject name for the certreq:

set subject-name *subject-name*

Step 14 Exit:

exit

Step 15 Set the modulus:

set modulus *modulus*

Step 16 Set the regeneration for the certificate request:

set regenerate *{ yes | no }*

Step 17 Set the trustpoint:

set trustpoint interca

Step 18 Exit:

exit

Step 19 Enter the newly created trustpoint:

enter trustpoint interca

Step 20 Generate certificate signing request:

set certchain

Example:

```
-----BEGIN CERTIFICATE-----
MIIF3TCCA8WgAwIBAgIBADANBgkqhkiG9w0BAQsFADBwMQswCQYDVQQGEwJVUzEL
MAkGA1UECAwCQ0ExDDAKBgNVBAcMA1NKQzEOMAwGA1UECgwFQ2lzY28xDTALBgNV
```

BAsMBFNUQlUxCzAJBgNVBAMMAkNBMRowGAYJKoZIhvcNAQkBFgtzc3BAc3NwLm5l
dDAeFw0xNjEyMDgxOTMzNTJaFw0yNjEyMDYxOTMzNTJaMHAxCzAJBgNVBAYTAlVT
MQswCQYDVQQIDAJDQTEMMAoGA1UEBwwDU0pDMQ4wDAYDVQQKDAVDaXNjbzENMAsG
A1UECwwEU1RCVTELMAkGA1UEAwwCQ0ExGjAYBgkqhkiG9w0BCQEWC3NzcEBzc3Au
bmV0MIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEA2ukWyMLQuLqTvhq7
zFb3Oz/iyDG/ui6mrLIYn8wE3E39XcXA1/x9IHCmxFKNJdD7EbsggfOuy0Bj+Y4s
+uZ1VapBXV/JrAie7bNn3ZYrI29yuyOrIqoi9k9gL/oRBzH18BwBwGHBOz3hGrSK
Yc2yhsq9y/6yI3nSuLZm6ybmUKjTa+B4YuhDTz4hl/I9x/J5nbGiab3vLDKss1nO
xP9+1+Lc690Vl8/mNPWdjCjDI+U/L9keYs/rbZdRSeXy9kMae42+4FlRHDJjPcSN
Yw1g/gcR2F7QUKRygKckJKXDX2QIiGYSctlSHj18O87o5s/pmQAWWRGkKpfDv3oH
cMPgI2T9rC0D8NNcgPXj9PFKfexoGNGwNTO85fK3kjgMOdWbdeMG3EihxEEOUPD0
Fdu0HrTM5lvwb+vr5wE9HsAiMJ8UuujmHqH5mlwyy3Me+cEDHo0hLeNs+AFrqEXQ
e9S+KZC/dq/9zOLpRsVqSfJsAuVl/QdPDbWShjflE/fP2Wj01PqXywQydzymVvgE
wEZaoFg+mlGJm0+q4RDvnpzEviOYNSAGmOkILh5HQ/eYDcxvd0qbORWb31H32ySl
Ila6UTT9+vnND1f838fxvNvr8nyGD2S/LVaxnZlO4jcSIvtdizbbT8u5B4VcLKIC
x0vkqjo6RvNZJ52sUaD9C3UodTUCAwEAAaOBgTB/MC8GA1UdHwQoMCYwJKAioCCG
Hmh0dHA6Ly8xOTIuMTY4LjQuMjkvcm9vdGNhLmNybDAdBgNVHQ4EFgQU7Jg01A74
jpx8U0APk76pVfYQQ5AwHwYDVR0jBBgwFoAU7Jg01A74jpx8U0APk76pVfYQQ5Aw
DAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQsFAAOCAgEAvI8ky2jiXc4wPiMuxIfY
W7DRmszPUWQ7edor7yxuCqzHLVFFOwYRudsyXbv7INR3rJ/X1cRQj9+KidWWVxpo
pFahRhzYxVZ10DHKIzGTQS3jiHgrF3Z8ohWbL15L7PEDlrxMBoJvabPeQRgTmY/n
XZJ7qRYbypO3gUMCaCZ12raJc3/DIpBQ29yweCbUkc9qiHKA0IbnvAxoroHWmBld
94LrJCggfMQTuNJQszJiVVsYJfZ+utlDp2QwfdDv7B0JkwTBjdwRSfotEbc5R18n
BNXYHqxuoNMmqbS3KjCLXcH6xIN8t+UkfP89hvJt/fluJ+s/VJSVZWK4tAWvR7wl
QngCKRJW6FYpzeyNBctiJ07wO+Wt4e3KhIjJDYvA9hFixWcVGDf2r6QW5BYbgGOK
DkHb/gdr/bcdLBKN/PtSJ+prSrpBSaA6rJX8D9UmfhqqN/3f+sS1fM4qWORJc6G2
gAcg7AjEQ/0do512vAI8p8idOg/Wv1O17mavzLpcue05cwMCX9fKxKZZ/+7Pk19Y
ZrXS6uMn/CGnViptn0w+uJ1IRj1oulk+/ZyPtBvFHUkFRnhoWj5SMFyds2IaatyI
47N2ViaZBxhU3GICaH+3O+8rs9Kkz9tBZDSnEJVZA6yxaNCVP1bRUO20G3oRTmSx
8iLBjN+BXggxMmG8ssHisgw=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFqDCCA5CgAwIBAgIBBDANBgkqhkiG9w0BAQsFADBwMQswCQYDVQQGEwJVUzEL
MAkGA1UECAwCQ0ExDDAKBgNVBAcMA1NKQzEOMAwGA1UECgwFQ2lzY28xDTALBgNV
BAsMBFNUQlUxCzAJBgNVBAMMAkNBMRowGAYJKoZIhvcNAQkBFgtzc3BAc3NwLm5l
dDAeFw0xNjEyMTUyMTM0NTRaFw0yNjEyMTMyMTM0NTRaMHwxCzAJBgNVBAYTAlVT
MQswCQYDVQQIDAJDQTEPMA0GA1UECgwGbmV3c3RnMRAwDgYDVQQLDAduZXdzdGJ1
MRMwEQYDVQQDDAppbnRlcm0xLWNhMSgwJgYJKoZIhvcNAQkBFhlpbnRlcm0xLWNh
QGludGVybTEtY2EubmV0MIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEA
wLpNnyEx5I4P8uDoWKWF3IZsegjhLANsodxuAUmhmwKekd0OpZZxHMw1wSO4IBX5
4itJS0xyXFzPmeptG3OXvNqCcsT+4BXl3DoGgPMULccc4NesHeg2z8+q3SPA6uZh
iseWNvKfnUjixbQEBtcrWBiSKnZuOz1cpuBn34gtgeFFoCEXN+EZVpPESiancDVh
8pCPlipc/08ZJ3o9GW2j0eHJN84sguIEDL812ROejQvpmfqGUq11stkIIuh+wB+V
VRhUBVG7pV57I6DHeeRp6cDMLXaM3iMTelhdShyo5YUaRJMak/t8kCqhtGXfuLlI
E2AkxKXeeveR9n6cpQd5JiNzCT/t9IQL/T/CCqMICRXLFpLCS9o5S5O2B6QFgcTZ
yKR6hsmwe22wpK8QI7/5oWNXlolb96hHJ7RPbG7RXYqmcLiXY/d2j9/RuNoPJawI
hLkfhoIdPA28xlnfIB1azCmMmdPcBO6cbUQfCj5hSmk3StVQKgJCjaujz55TGGd1
GjnxDMX9twwz7Ee51895Xmtr24qqaCXJoW/dPhcIIXRdJPMsTJ4yPG0BieuRwd0p
i8w/rFwbHzv4C9Fthw1JrRxH1yeHJHrLlZgJ5txSaVUIgrgVCJaf6/jrRRWoRJwt
AzvnzYql2dZPCcEAYgP7JcaQpvdpuDgq++NgBtygiqECAwEAAaNBMD8wDAYDVR0T
BAUwAwEB/zAvBgNVHR8EKDAmMCSgIqAghh5odHRwOi8vMTkyLjE2OC40LjI5L2lu
dGVybS5jcmwwDQYJKoZIhvcNAQELBQADggIBAG/XujJh5G5UWo+cwTSitAezWbJA
h1dAiXZ/OYWZSxkFRliErKdupLqL0ThjnX/wRFfEXbrBQwm5kWAUUDr97D1Uz+2A
8LC5I8SWKXmyf0jUtsnEQbDZb33oVL7yXJk/A0SF0jihpPheMA+YRazalT9xj9KH
PE7nHCJMbb2ptrHUyvBrKSYrSeEqOpQU2+otnFyV3rS9aelgVjuaWyaWOc3lZ1Oi
CC2tJvY3NnM56j5iesxUCeY/SZ2/ECXN7RRBViLHmA3gFKmWf3xeNiKkxmJCxOaa
UWPC1x2V66I8DG9uUzlWyd79O2dy52aAphAHC6hqlzb6v+gw1Tld7UxaqVd8CD5W

```
ATjNs+ifkJS1h5ERxHjgcurZXOpR+NWpwF+UDzbMXxx+KAAXCI6ltCd8Pb3wOUC3
PKvwEXaIcCcxGx71eRLpWPZFyEoi4N2NGE9OXRjz0K/KERZgNhsIW3bQMjcw3aX6
OXskEuKgsayctnWyxVqNnqvpuz06kqyubh4+ZgGKZ5LNEXYmGNz3oED1rUN636Tw
SjGAPHgeROzyTFDixCei6aROlGdP/Hwvb0/+uThIe89g8WZ0djTKFUM8uBO3f+II
/cbuyBO1+JrDMq8NkAjxKlJlp1c3WbfCue/qcwtcfUBYZ4i53a56UNF5Ef0rpy/8
B/+07Me/p2y9Luqa
-----END CERTIFICATE-----
ENDOFBUF
```

Step 21 Show the certificate signing request:

show certreq

Example:

```
Firepower-chassis# /security/keyring # show certreq
Certificate request subject name: SSP
Certificate request ip address: 192.168.0.111
Certificate request FI A ip address: 0.0.0.0
Certificate request FI B ip address: 0.0.0.0
Certificate request e-mail name:
Certificate request ipv6 address: ::
Certificate request FI A ipv6 address: ::
Certificate request FI B ipv6 address: ::
Certificate request country name: US
State, province or county (full name): CA
Locality name (eg, city): SJC
Organisation name (eg, company): Cisco
Organisational Unit Name (eg, section): Sec
DNS name (subject alternative name):
Request:
-----BEGIN CERTIFICATE REQUEST-----
MIICwTCCAakCAQAwVTELMAkGA1UEBhMCVVMxCzAJBgNVBAgMAkNBMQwwCgYDVQQH
DANTSkMxDjAMBgNVBAoMBUNpc2NvMQ0wCwYDVQQLDARTVEJVMQwwCgYDVQQDDANT
U1AwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDq292Rq3t0laoxPbfE
p/lTKr6rxFhPqSSbtm6sXer//VZFiDTWODockDItuf4Kja215mIS0RyvEYVeRgAs
wbN459wm0BASd8xCjIhsuHDV7yHu539BnvRW6Q2o+gHeSRwckqjClK/tsIxsPkV0
6OduZYXk2bnsLWs6tNk3uzOIT2Q0FcZ1ET66C8fyyKWTrmvcZjDjkMm2nDFsPlX9
39TYPItDkJE3PocqyaCqmT4uobOuvQeLJh/efkBvwhb4BF8vwzRpHWTdjjU5YnR1
qiR4q7j1RmzVFxCDY3IVP/KDBoa5NyCLEUZECP5QCQFDzIRETZwVOKtxUVG0Nljd
K5TxAgMBAAGgJzAlBgkqhkiG9w0BCQ4xGDAWMBQGA1UdEQQNMAuCA1NTUIcEwKgA
rjANBgkqhkiG9w0BAQsFAAOCAQEARtRBoInxXkBYNlVeEoFCqKttu3+Hc7UdyoRM
2L2pjx5OHbQICC+8NRVRMYujTnp67BWuUZZl03dGP4/lbN6bC9P3CvkZdKUsJkN0
m1Ye9dgz7MO/KEcosarmoMl9WB8LlweVdt6ycSdJzs9shOxwT6TAZPwL7gq/1ShF
RJh6sq5W9p6E0SjYefK62E7MatRjDjS8DXoxj6gfn9DqK15iVpkK2QqT5rneSGj+
R+20TcUnT0h/S5K/bySEM/3U1gFxQCOzbzPuHkj28kXAVczmTxXEkJBFLVduWNo6
DT3u0xImiPR1sqW1jpMwbhC+ZGDtvgKjKHToagup9+8R9IMcBQ==
-----END CERTIFICATE REQUEST-----
```

Step 22 Enter IPSec mode:

scope ipsec

Step 23 Set the log verbose level:

set log-level *log_level*

Step 24 Create and enter an IPSec connection:

enter connection *connection_name*

Step 25 Set IPSec mode to tunnel or transport:

set mode *tunnel_or_transport*

Step 26 Set the local IP address:

set local-addr *ip_address*

Step 27 Set the remote IP address:

set remote-addr *ip_address*

Step 28 If using tunnel mode, set the remote subnet:

set remote-subnet *ip/mask*

Step 29 (Optional) Set the remote identity:

set remote-ike-ident *remote_identity_name*

Step 30 Set the keyring name:

set keyring-name *name*

Step 31 (Optional) Set the keyring password:

set keyring-passwd *passphrase*

Step 32 (Optional) Set the IKE-SA lifetime in minutes:

set ike-rekey-time *minutes*

The *minutes* value can be any integer between 60-1440, inclusive.

Step 33 (Optional) Set the Child SA lifetime in minutes (30-480):

set esp-rekey-time *minutes*

The *minutes* value can be any integer between 30-480, inclusive.

Step 34 (Optional) Set the number of retransmission sequences to perform during initial connect:

set keyringtries *retry_number*

The *retry_number* value can be any integer between 1-5, inclusive.

Step 35 (Optional) Enable or disable the certificate revocation list check:

set revoke-policy *{ relaxed | strict }*

Step 36 Enable the connection:

set admin-state enable

Step 37 Reload connections:

reload-conns

The system stops all connections and then reloads them. All connections will try to re-establish.

Step 38 (Optional) Add the existing trustpoint name to IPsec:

create authority *trustpoint_name*

Step 39 Configure the enforcement of matching cryptographic key strength between IKE and SA connections:

set sa-strength-enforcement *yes_or_no*

Configure Static CRL for a Trustpoint

Revoked certifications are kept in the Certification Revocation List (CRL). Client applications use the CRL to check the authentication of a server. Server applications utilize the CRL to grant or deny access requests from client applications which are no longer trusted.

You can configure your Firepower 4100/9300 chassis to validate peer certificates using Certification Revocation List (CRL) information. This option is one of a number offered for achieving Common Criteria certification compliance on your system. For more information, see Security Certifications Compliance, on page 1.

Perform these steps to validate peer certificates using CRL information.

Procedure

Step 1 From the FXOS CLI, enter security mode:

scope security

Step 2 Enter trustpoint mode:

scope trustpoint *trustname*

Step 3 Enter revoke mode:

scope revoke

Step 4 Download the CRL file(s):

import crl *protocol*://*user_id@CA_or_CRL_issuer_IP*/tmp/*DoDCA1CRL1.crl*

Note DER format static CRL is not supported in FXOS. You must convert the DER format CRL file to PEM format using the following command:

```
openssl crl -in filename.crl -inform DER -outform PEM -out crl.pem
```

Step 5 (Optional) Show the status of the import process of CRL information:

show import-task detail

Step 6 Set the certificate revocation method to CRL-only:

set certrevokemethod {crl}

About the Certificate Revocation List Check

You can configure your Certificate Revocation List (CRL) check mode to be either strict or relaxed in IPSec and secure LDAP connections.

FXOS harvests dynamic (non-static) CRL information from the CDP information of an X.509 certificate, which indicates dynamic CRL information. System administration downloads static CRL information manually, which indicates local CRL information in the FXOS system. FXOS processes dynamic CRL information against the current processing certificate in the certificate chain. The static CRL is applied to the whole peer certificate chain.

For steps to enable or disable certificate revocation checks for your secure LDAP and IPSec connections, see Configure IPSec Secure Channel, on page 3 and Creating an LDAP Provider.

Note

- If the Certificate Revocation Check Mode is set to Strict, static CRL is only applicable when the peer certificate chain has a level of 1 or higher. (For example, when the peer certificate chain contains only the root CA certificate and the peer certificate signed by the root CA.)
- When configuring static CRL for IPSec, the Authority Key Identifier (authkey) field must be present in the imported CRL file. Otherwise, IPSec considers it invalid.
- Static CRL takes precedence over Dynamic CRL from the same issuer. When FXOS validates the peer certificate, if a valid (determined) static CRL of the same issuer exists, FXOS ignores the CDP in the peer certificate.
- Strict CRL checking is enabled by default in the following scenarios:
  - Newly created secure LDAP provider connections, IPSec connections, or Client Certificate entries
  - Newly deployed FXOS chassis managers (deployed with an initial starting version of FXOS 2.3.1.x or later)

The following tables describe the connection results, depending on your certificate revocation list check setting and certificate validation.

Table 1: Certificate Revocation Check Mode set to Strict without a local static CRL

Without local static CRL	LDAP Connection	IPSec Connection
Checking peer certificate chain	Full certificate chain is required	Full certificate chain is required
Checking CDP in peer certificate chain	Full certificate chain is required	Full certificate chain is required
CDP checking for Root CA certificate of the peer certificate chain	Yes	Not applicable
Any certificate validation failure in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message

Without local static CRL	LDAP Connection	IPSec Connection
Any certificate revoked in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
One CDP is missing in the peer certificate chain	Connection fails with syslog message	Peer certificate: connection fails with syslog message Intermediate CAs: connection succeeded
One CDP CRL is empty in the peer certificate chain with valid signature	Connection fails with syslog message	Connection succeeds
Any CDP in the peer certificate chain cannot be downloaded	Connection fails with syslog message	Peer certificate: Connection fails with syslog message Intermediate CA: connection succeeded
Certificate has CDP, but the CDP server is down	Connection fails with syslog message	Peer certificate: Connection fails with syslog message Intermediate CA: connection succeeded
Certificate has CDP, server is up, and CRL is on CDP, but the CRL has an invalid signature	Connection fails with syslog message	Peer certificate: Connection fails with syslog message Intermediate CA: connection succeeded

Table 2: Certificate Revocation Check Mode set to Strict with a local static CRL

With local static CRL	LDAP Connection	IPSec Connection
Checking peer certificate chain	Full certificate chain is required	Full certificate chain is required
Checking CDP in peer certificate chain	Full certificate chain is required	Full certificate chain is required
CDP checking for Root CA certificate of the peer certificate chain	Yes	Not applicable
Any certificate validation failure in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
Any certificate revoked in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
One CDP is missing in the peer certificate chain (Certificate Chain level is 1)	Connection succeeds	Connection succeeds

With local static CRL	LDAP Connection	IPSec Connection
One CDP CRL is empty in the peer certificate chain (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Any CDP in the peer certificate chain cannot be downloaded (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Certificate has CDP, but the CDP server is down (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Certificate has CDP, server is up, and CRL is on CDP, but the CRL has an invalid signature (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Peer Certificate Chain level is higher than 1	Connection fails with syslog message	If combined with CDP, connection succeeds If there is no CDP, connection fails with syslog message

Table 3: Certificate Revocation Check Mode set to Relaxed without a local static CRL

Without local static CRL	LDAP Connection	IPSec Connection
Checking peer certificate chain	Full certificate chain	Full certificate chain
Checking CDP in the peer certificate chain	Full certificate chain	Full certificate chain
CDP checking for Root CA certificate of the peer certificate chain	Yes	Not applicable
Any certificate validation failure in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
Any certificate revoked in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
One CDP is missing in the peer certificate chain	Connection succeeds	Connection succeeds
One CDP CRL is empty in the peer certificate chain	Connection succeeds	Connection succeeds
Any CDP in the peer certificate chain cannot be downloaded	Connection succeeds	Connection succeeds
Certificate has CDP, but the CDP server is down	Connection succeeds	Connection succeeds

Without local static CRL	LDAP Connection	IPSec Connection
Certificate has CDP, server is up, and CRL is on CDP, but the CRL has an invalid signature	Connection succeeds	Connection succeeds

Table 4: Certificate Revocation Check Mode set to Relaxed with a local static CRL

With local static CRL	LDAP Connection	IPSec Connection
Checking peer certificate chain	Full certificate chain	Full certificate chain
Checking CDP in the peer certificate chain	Full certificate chain	Full certificate chain
CDP checking for Root CA certificate of the peer certificate chain	Yes	Not applicable
Any certificate validation failure in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
Any certificate revoked in the peer certificate chain	Connection fails with syslog message	Connection fails with syslog message
One CDP is missing in the peer certificate chain (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
One CDP CRL is empty in the peer certificate chain (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Any CDP in the peer certificate chain cannot be downloaded (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Certificate has CDP, but the CDP server is down (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Certificate has CDP, server is up, and CRL is on CDP, but the CRL has an invalid signature (Certificate Chain level is 1)	Connection succeeds	Connection succeeds
Peer Certificate Chain level is higher than 1	Connection fails with syslog message	If combined with CDP, connection succeeds If there is no CDP, connection fails with syslog message

Configure CRL Periodic Download

You can configure your system to periodically download a (CRL) so that a new CRL is used every 1 to 24 hours to validate certificates.

You can use the following protocols and interfaces with this feature:

- FTP
- SCP
- SFTP
- TFTP
- USB

Note

- SCEP and OCSP are not supported.
- You can only configure one periodic download per CRL.
- One CRL is supported per trustpoint.

Note You can only configure the period in one-hour intervals.

Perform these steps to configure CRL periodic download.

Before you begin

Ensure that you have already configured your Firepower 4100/9300 chassis to validate peer certificates using (CRL) information. For more information, see Configure Static CRL for a Trustpoint, on page 9.

Procedure

Step 1 From the FXOS CLI, enter security mode:

scope security

Step 2 Enter trustpoint mode:

scope trustpoint

Step 3 Enter revoke mode:

scope revoke

Step 4 Edit the revoke configuration:

sh config

Step 5 Set your preferred configuration:

Example:

```
set certrevokemethod crl
set crl-poll-filename rootCA.crl
set crl-poll-path /users/myname
set crl-poll-period 1
set crl-poll-port 0
set crl-poll-protocol scp
! set crl-poll-pwd
set crl-poll-server 182.23.33.113
set crl-poll-user myname
```

Step 6 Exit the configuration file:

exit

Step 7 (Optional) Test the new configuration by downloading a new CRL:

Example:

```
Firepower-chassis /security/trustpoint/revoke # sh import-task

Import task:
File Name Protocol Server          Port      Userid        State
--------- -------- ---------------- ----------- -------------- -----
rootCA.crl Scp     182.23.33.113    0         myname        Downloading
```

Set the LDAP Key Ring Certificate

You can configure a secure LDAP client key ring certificate to support a TLS connection on your Firepower 4100/9300 chassis. This option is one of a number offered for achieving Common Criteria certification compliance on your system. For more information, see Security Certifications Compliance, on page 1.

Note If Common Criteria mode is enabled, you must have SSL enabled, and you must use the server DNS information to create the key ring certificate.

If SSL is enabled for the LDAP server entry, key ring information is referenced and checked when forming a connection.

LDAP server information has to be DNS information in the CC mode for the secure LDAP connection (with SSL enabled).

Perform these steps to configure a secure LDAP client key ring certificate:.

Procedure

Step 1 From the FXOS CLI, enter security mode:

scope security

Step 2 Enter LDAP mode:

scope ldap

Step 3 Enter LDAP server mode:

enter server *{server_ip|server_dns}*

Step 4 Set the LDAP key ring:

set keyring *keyring_name*

Step 5 Commit the configuration:

commit-buffer